

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

Факультет електроніки
Кафедра електронних приладів та пристроїв

До захисту допущено

Завідувач кафедри, проф., д.т.н.

_____ Л.Д.Писаренко

« ____ » _____ 2019 р.

ДИПЛОМНА РОБОТА

на здобуття ступеня бакалавра

з напрямку підготовки **6.050802 – Електронні пристрої та системи**

на тему **«Розробка електронного стенду на основі мікроконтролера для
налагодження програм»**

Виконав:

Студент 3 курсу, гр. ДЕ-пб1 Аршан Євгеній Віталійович _____

Керівник:

Ст.викл., к.т.н., с.н.с. Семікіна Т.В. _____

Нормоконтроль:

Доцент, к.т.н. Чадюк В.О. _____

Рецензент:

Доцент кафедри

К.т.н., с.н.с. Бобренко Ю.М. _____

Засвідчую, що у цій бакалаврській роботі
немає запозичень з праць інших авторів без
відповідних посилань.

Студент _____
(підпис)

Київ-2019

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»**

Факультет електроніки

Кафедра електронних приладів та пристроїв

Рівень вищої освіти – перший (бакалаврський)

Напрямок підготовки (програма професійного спрямування) – 6.050802 –

Електронні пристрої та системи (Електронні прилади та пристрої) ІСпальність о

ЗАТВЕРДЖУЮ

Завідувач кафедри, проф., д.т.н.

_____ Л.Д.Писаренко

« ____ » _____ 2019 р.

ЗАВДАННЯ

на дипломну роботу студенту

Аршану Євгенію Віталійовичу

1. Тема роботи «Розробка електронного стенду на основі мікроконтролера для налагодження програм»

і керівник роботи Семікіна Тетяна Вікторівна, ст.викл., к.т.н., с.н.с.

затверджені наказом по університету від « ____ » _____ 2019 р., № 1404-с

2. Термін подання студентом роботи : « ____ » _____ 2019 р.

3. Вихідні дані до роботи: Розробка електронного стенду на основі мікроконтролера для налагодження програм. Розробка схеми електричної принципової та креслення друкованої плати. Написання програми для перевірки роботи стенду.

4. Зміст роботи: Анотація; вступ; огляд науково-технічної літератури по мікропроцесорам та мікроконтролерам; огляд існуючих прототипів налагоджувальних плат; розробка технічного завдання та принципової схеми електронного стенду; розробка конструкторсько-технічної документації.

5. Перелік ілюстративного матеріалу: Схема електрична принципова електронного стенду на основі мікроконтролера для налагодження програм, креслення друкованої плати, алгоритм розробленої програми, функціональна схема стенду.

6. Дата видачі завдання : _____ р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів виконання дипломної роботи	Термін виконання етапів роботи	Примітка
1	Огляд науково-технічної літератури по мікроконтролерам та мікропроцесорам.	01.10.2018	
2	Аналіз схем та принципу роботи існуючих аналогів налагоджувальних плат на основі мікроконтролерів	14.12.2018	
3	Розробка функціональної схеми електронного стенду на основі мікроконтролера для налагодження програм	10.02.2019	
4	Розробка електричної принципової схеми та креслення друкованої плати стенду	15.03.2019	
5	Розробка друкованої плати електронного стенду на основі мікроконтролера для налагодження програм	05.04.2019	
6	Написання програми та алгоритму	25.04.2019	
7	Оформлення графічної частини, пояснювальної записки, плакатів, підготовка доповіді	25.05.2019	

Студент гр. ДЕ-пб1

Є.В.Аршан

Керівник роботи

Т.В.Семікіна

[illegible]

ПОЯСНЮВАЛЬНА ЗАПИСКА

до дипломної роботи

**на тему: «Розробка електронного стенду на основі
мікроконтролера для налагодження програм»**

РЕФЕРАТ

Розробка електронного стенду на основі мікроконтролера для налагодження програм

Дипломна робота освітньо-кваліфікаційного рівня «Бакалавр» спеціальності 171- Електроніка, спеціалізації – Електронні прилади та пристрої. Аршан Євгеній Віталійович. КПІ ім. Ігоря Сікорського. Факультет електроніки, кафедра «Електронні прилади та пристрої». Група ДЕ-пб1. – К.: КПІ ім. Ігоря Сікорського, 2019. – 90 с., іл. 16, табл. 1., вик. джерел 21.

Ключові слова: мікроконтролер ATmega32A, Splan, Sprint–Layout, конструкторська документація, налагоджувальна плата.

Короткий зміст роботи: В дипломній роботі представлено огляд науково-технічної літератури по мікропроцесорам та мікроконтролерам.

При огляді літератури було досліджено два методи налагодження роботи програм для мікроконтролерних пристроїв, досліджено застосування цих методів на практиці, та проблеми що можуть виникати при написанні програм.

В якості рішення цих проблем в дипломному проєкті була запропонована розробка електронного стенду (налагоджувальної плати) для налагодження роботи програм. Завдяки застосуванню налагоджувальної плати, коректність роботи програм можна перевірити на відповідних блоках плати, а вбудовані системи діагностики і прошивки дозволяють записати нову прошивку в мікроконтролер одним рухом.

В дипломній роботі була вирішена задача розробки електронного стенду на основі мікроконтролера для подальшого налагодження роботи програм. Програма зв'язує мікроконтролер з виконавчими елементами. Робота таких програм перевіряється за допомогою електронного стенду, котрий підключається безпосередньо до підсистеми. Розроблений та представлений в цій роботі електронний стенд відрізняється від вже існуючих аналогів тим, що стандартна комплектація не вміщує в себе нічого зайвого, що обумовлюватиме низьку ціну і високу практичну цінність. Мультизадачність електронного стенду обумовлена можливістю підключення додаткових

підсистем, що потрібні для візуальної перевірки роботи конкретної програми, в вигляді периферійних пристроїв, до вільних портів мікроконтролера.

Стенд можна використовувати для лабораторних робіт по вивченню роботи мікроконтролерів.

В роботі було розроблено конструкторську документацію необхідну для виготовлення стенду. Для цього було застосовано програми Splan та Sprint–Layout.

При розробці друкованої плати було використано програму Sprint–Layout.

Було розроблено креслення друкованої плати: зображення друкованої плати з боку друкованого монтажу; розміри, граничні відхилення всіх її елементів (отворів, провідників), а також розміри відстаней між ними; необхідні технічні вимоги, відомості про матеріал.

Після створення схеми електричної принципової, переліку елементів та креслення друкованої плати було зроблено монтаж пристрою.

При розробці конструкції електронного стенду було спроектовано схеми стенду на основі мікроконтролера АТМega32А, було вибрано радіоелементи, що забезпечують надійну і стійку роботу схеми, було досягнуте оптимальне співвідношення ціни та якості вибраних електро-радіо елементів (ЕРЕ).

Було написано базову програму для перевірки коректності роботи функціональних блоків електронного стенду. Для кращого розуміння кроків виконання програми було створено алгоритм програми.

АНОТАЦІЯ

В дипломній роботі представлено огляд науково-технічної літератури по мікропроцесорам та мікроконтролерам.

В дипломній роботі представлено конструкторську документацію електронного стенду на основі мікроконтролера. Даний стенд розроблений для візуальної перевірки коректної роботи програм, написаних для виконання певних дій електронними пристроями, зняття показників датчиків, виконання світлової індикації. Електронний стенд дає можливість підключення додаткових зовнішніх периферійних пристроїв для виконання конкретних задач.

Було написано базову програму для перевірки коректності роботи функціональних блоків електронного стенду. Для кращого розуміння кроків виконання програми було створено алгоритм програми.

S U M M A R Y

The diploma project presents a review of scientific and technical literature on microprocessors and microcontrollers.

The diploma project presents the design documentation for an electronic stand based on a microcontroller. This stand is designed for visual verification of the correct operation of programs written for certain actions by electronic devices, the removal of indicators of sensors, the performance of the light indication. An electronic stand enables the connection of additional external peripherals to perform specific tasks.

It was written a basic program for checking the correctness of the work of the functional blocks of the electronic stand. For a better understanding of the program implementation steps, an algorithm was created.

ЗМІСТ

ВСТУП.....	13
1 ОГЛЯД НАУКОВО-ТЕХНІЧНОЇ ЛІТЕРАТУРИ	16
1.1 Мікропроцесори	16
1.1.1 Прогрес мікропроцесорів Intel.....	16
1.1.2 Значення розміру процесу процесора	19
1.1.3 Принцип роботи МП.....	22
1.2 Мікроконтролери	25
1.2.1 Типи мікроконтролерів.....	27
1.2.2 Регістри спеціальних функцій	29
1.3 Відмінності між мікропроцесором і мікроконтролером	30
1.4 ATmega32 - 8-бітний мікроконтролер AVR	33
1.5 Огляд існуючих прототипів	35
Висновки до розділу 1	39
2. РОЗРОБКА ТА ОПИС СТЕНДУ	40
2.1 Обґрунтування схемних рішень та використаної елементної бази....	40
2.2 Опис і робота стенду.....	41
2.2.1 Призначення стенду.....	41
2.2.2 Характеристики стенду	43
2.3 Склад стенду, призначення складових частин	45
2.3.1 Будова та робота.....	46
Висновки до розділу 2	48
3 РОЗРОБКА КОНСТРУКТОРСЬКО-ТЕХНІЧНОЇ ДОКУМЕНТАЦІЇ	49
3.1 Програмні середовища	49

					БР.171.061.001 ПЗ								
Змн.	Арк.	№ докум.	Підпис	Дата									
Розроб.		Аршан Є.В.			Розробка електронного стенду на основі мікроконтролера для налагодження програм. Пояснювальна записка				Літ.	Арк.	Аркушів		
Перевір.		Семікіна Т.В.									10	90	
Реценз.		Бобренко Ю.М							«КПІ ім. Ігоря Сікорського», ФЕЛ, ЕПП, гр. ДЕ-п61				
Н. Контр.		Чадюк В.О.											
Затверд.		Писаренко Л.Д.											

3.2. Розробка складального креслення та плати друкованої електронного стенду на основі мікроконтролера для налагодження програм.....	49
Висновки до розділу 3	55
4 ЕКСПЕРИМЕНТАЛЬНА ЧАСТИНА: Технологічний процес виготовлення плати електронного стенду на основі мікроконтролера для налагодження програм.....	56
Висновки до розділу 4	59
5 РОЗРАХУНКОВА ЧАСТИНА.....	60
5.1 Мова програмування Assembler	60
5.1.1 Переваги і недоліки.....	61
5.2 Середовище програмування.....	61
5.3 Опис розробленого алгоритму базової програми електронного стенду.....	63
Висновки до розділу 5	66
ВИСНОВКИ.....	67
ПЕРЕЛІК ВИКОРИСТАНОЇ НАУКОВО-ТЕХНІЧНОЇ ЛІТЕРАТУРИ.....	69
Додаток А. Технічне завдання	
Додаток Б. Схема електрична принципова	
Додаток В. Специфікація	
Додаток Г. Креслення друкованої плати	
Додаток Д. Лістинг програми	

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

ЦП – Центральний процесор

ПК – Персональний комп'ютер

MIPS – Мільйони інструкцій в секунду (показник продуктивності)

АЛП – Арифметико-логічний пристрій.

МП – Мікропроцесор

RD /WR – Рядок читання/ запису

РК-дисплеї – Рідкокристалічний дисплей

МК – Мікроконтролер

RAM – Оперативна пам'ять з довільним доступом

ROM – Постійний запам'ятовуючий пристрій

EPROM – Програмована пам'ять тільки для читання з можливістю
стирання

EEPROM – Програмована пам'ять тільки для читання з можливістю
електричного стирання

RISC – комп'ютер зі скороченим набором інструкцій

CISC – комп'ютер з комплексним набором інструкцій

ОЗП – Оперативно запам'ятовуючий пристрій

ПЗП – Постійний запам'ятовуючий пристрій

SFR – Регістри спеціальних функцій

CMOS (КМОП) – Комплементарна структура метал-оксид-напівпровідник

SPI – Послідовний периферійний інтерфейс

АЦП – Арифметико цифровий перетворювач

МПС – Мікропроцесорна система

ЕС – Електронний стенд

ЕРЕ – Електро-радіо елементи

ШИМ – Широтно-імпульсна модуляція

IDP – інтегрована платформа розробки

					БР.171.061.001 ПЗ	Арк.
						12
Змн.	Арк.	№ докум.	Підпис	Дата		

ВСТУП

Мікропроцесори та похідні від них – мікроконтролери – є поширеним елементом інфраструктури сучасного суспільства, заснованого на електроніці і комунікаціях.

За останні роки мікроконтролери AVR придбали велику популярність, залучаючи розробників досить вигідним співвідношенням «ціна/швидкодія/енергоспоживання», зручними режимами програмування, доступністю програмно-апаратних засобів підтримки і широкою номенклатурою випущених кристалів. Мікроконтролери (МК) цієї серії є чудовим інструментом для створення сучасних високопродуктивних і економічних контролерів багатоцільового призначення [1].

Сучасна стратегія модульного проектування забезпечує споживача різноманітністю моделей МК з одним і тим же процесорним ядром. Така структурна і різноманітність відкриває перед розробником можливість вибору оптимального МК. Програмування мікроконтролерів здійснюється різними шляхами:

1. Шляхом тестування на симуляторах;
2. Розробка макетів.

Але в цих двох методах є суттєві недоліки: робота в симуляторі часто відрізняється від роботи з електронними пристроями, при збиранні на макетній платі можна зробити помилку і вивести контролер з ладу або підключити невірно.

Відмінним рішенням для усунення цих проблем є налагоджувальна плата. Це плата, де вже все правильно підключено і розведено. Функціонування електронних пристроїв можна перевірити на еталонній демопрограмі, а вбудовані системи діагностики і прошивки дозволяють записати нову прошивку в мікроконтролер одним рухом. Залишається тільки експериментувати, не відволікаючись на сторонні фактори.

					БР.171.061.001 ПЗ	Арк.
						13
Змн.	Арк.	№ докум.	Підпис	Дата		

Також налагоджувальна плата служить зручним засобом для швидкого створення прототипів та обкатки вузлів, перевірки ідей і методів. Зібрати на демоплаті, перевірити, що ідея працює, як потрібно відкоригувати, а після вже налагоджений код використати в реальному проекті.

Світові фірми пропонують широкий вибір налагоджувальних плат, які мають більше можливостей, але вартість таких плат є дуже високою. Найбільш відомими брендами в області виробництва налагоджувальних плат і модулів розширення є наступні компанії: Atmel, Digilent, Embest, Freescale, Future Technology, IAR Systems, Microchip, mikroElektronika, Olimex, Seeed Studio, ST Microelectronics, ST Microelectronics, Texas Instruments, WaveShare. Заводські прототипи налагоджувальних плат та електронних стендів на основі мікроконтролера для налагодження програм є електронними пристроями для візуальної перевірки коректності роботи програм. Але такі електронні стенди виходять на ринок продажу в стандартних комплектаціях в залежності від рівня складності системи. Це обумовлює високу ціну так як для виконання різних типів завдань потрібно купляти новий електронний стенд який буде відповідати вимогам певного проекту для якого розроблене конкретне програмне забезпечення яке потрібно перевірити на коректність.

При всій різниці у найменуваннях, всі налагоджувальні плати виконують одну функцію: зменшують ймовірність непрацездатності програми через неправильне функціонування апаратної частини проекту – «заліза». Розробник як би відсувається від апаратної частини проекту, і концентрує свої зусилля на процесі написання і налагодження програми.

В даній роботі вирішується задача розробки електронного стенду на основі мікроконтролера для подальшого налагодження роботи програм. Програма зв'язує мікроконтролер з певними виконавчими елементами, наприклад: таймер, світлофор, світлова індикація. Робота таких програм перевіряється за допомогою електронного стенду, котрий підключається безпосередньо до підсистеми. Розроблений та представлений в цій роботі електронний стенд відрізняється від вже існуючих аналогів тим, що стандартна комплектація не вміщує в себе нічого

					БР.171.061.001 ПЗ	Арк.
						14
Змн.	Арк.	№ докум.	Підпис	Дата		

зайвого, що обумовлюватиме низьку ціну і високу практичну цінність. Мультизадачність електронного стенду обумовлена можливістю підключення додаткових підсистем, що потрібні для візуальної перевірки роботи конкретної програми, в вигляді периферійних пристроїв, до вільних портів мікроконтролера.

Стенд можна використовувати для лабораторних робіт по вивченню роботи мікроконтролерів.

					БР.171.061.001 ПЗ	Арк.
						15
Змн.	Арк.	№ докум.	Підпис	Дата		

1 ОГЛЯД НАУКОВО-ТЕХНІЧНОЇ ЛІТЕРАТУРИ

1.1 Мікропроцесори

Мікропроцесори, які використовуються, можуть бути різних типів та виготовлятися різними фірмами Pentium, K6, PowerPC, Sparc або будь-яким з багатьох інших брендів і типів мікропроцесорів, але всі вони працюють приблизно однаково й виконують однакові функції.

Мікропроцесор – також відомий як центральний процесор (ЦП) – являє собою повний обчислювальний механізм, виготовлений на одному кристалі. Першим мікропроцесором був Intel 4004, представлений в 1971 році. Intel 4004 був недостатньо потужним функціонал даного процесору дозволяв виконувати операції додавати і віднімання, продуктивність складала 4 біти за цикл [2]. Новим досягненням в виробництві мікропроцесорів стало те, що всі елементи містилися на одному чіпі. До Intel 4004 комп'ютери збиралися або з наборів мікросхем, або з окремих компонентів (транзистори підключалися по одному). 4004 –й працював на одному з перших портативних електронних калькуляторів.

1.1.1 Прогрес мікропроцесорів Intel

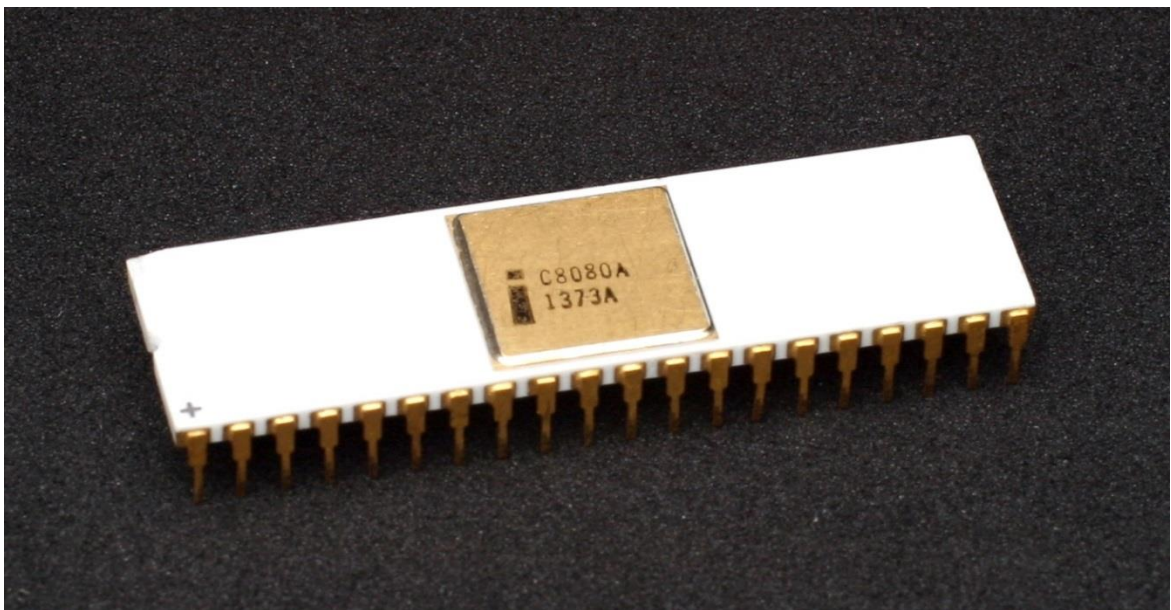


Рисунок 1.1. Intel 8080. Перший мікропроцесор в домашньому комп'ютері.

					БР.171.061.001 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		16

Першим мікропроцесором, що був представлений в домашньому комп'ютері, був Intel 8080 (рис. 1.1), повноцінний 8-розрядний комп'ютер на одному чіпі, представлений в 1974 році. Першим мікропроцесором, який викликав справжній сплеск на ринку, був Intel 8088, що був представлений в 1979 році і включений в комплект поставки. В IBM PC (який вперше з'явився близько 1982 року). В персональних комп'ютерах (ПК) поступово використовувалися більш прогресивні процесори, наприклад з 8088 було здійснено перехід на 80286, на 80386, на 80486, на Pentium, на Pentium II, на Pentium III, на Pentium 4. Всі ці Мікропроцесори виготовлялися компанією Intel. Виробництво було засноване на покращенні продуктивності і комплектації базової конструкції 8088. Pentium 4 може виконувати будь-який фрагмент коду, який працював на оригінальному 8088, але продуктивність збільшилася приблизно в 5000 разів. З 2004 року компанія Intel представила мікропроцесори з декількома ядрами і мільйонами транзисторів. Ці мікропроцесори слідують тим же загальним правилам, що і більш ранні чіпи [3].

Таблиця 1. Праметри процесорів розроблених компанією Intel (1974-2004 роки).

Назва	Дата випуску	Кількість транзисторів	Товщина, мкм	Тактова частота	Розмір даних	Продуктивність(MIPS)
8080	1974	6,000	6	2 МГц	8 біт	0.64
8088	1979	29,000	3	5 МГц	16 біт 8 бітна шина	0.33
80286	1982	134,000	1.5	6 МГц	16 біт	1
80386	1985	275,000	1.5	16 МГц	32 біти	5
80486	1986	1,200,000	1	25 МГц	32 біти	20
Pentium	1993	3,100,000	0.8	60 МГц	32 біти 64-бітна шина	100
Pentium II	1997	7,500,000	0.35	233 МГц	32 біти 64-бітна шина	≈300
Pentium III	1999	9,500,000	0.25	450 МГц	32 біти 64-бітна шина	≈510

Назва	Дата випуску	Кількість транзисторів	Товщина, мкм	Тактова частота	Розмір даних	Продуктивність (MIPS)
Pentium 4	2000	42,000,000	0.18	1.5 ГГц	32 біти 64-бітна шина	≈1,700
Pentium 4 “Prescott”	2004	125,000,000	0.09	3.6 ГГц	32 біти 64-бітна шина	≈7,000

Додаткова інформація по таблиці 1:

- **Дата випуску** - це рік, коли процесор був вперше представлений. Більшість процесорів повторно представлені на вищих тактових частотах протягом багатьох років після початкової дати випуску.
- **Кількість транзисторів** - це кількість транзисторів в чіпі. З розвитком технологій виробництва, кількість транзисторів на одному кристалі збільшується.
- **Товщина** - це товщина найменшого дроту на чіпі в мікронах. У міру зменшення розміру елемента на чіпі кількість транзисторів збільшується.
- **Тактова частота** - це максимальна частота, з якою чіп може бути синхронізований.
- **Розмір даних** - це ширина арифметико-логічного пристрою (АЛП). 8-бітний АЛП може додавати / віднімати / множити / і т.д. два 8-розрядних числа, в той час як 32-розрядний АЛП може маніпулювати 32-розрядними числами. 8-бітний АЛП повинен буде виконати чотири інструкції, щоб додати два 32-бітних числа, в той час як 32-бітний АЛП робить це в одній інструкції. У багатьох випадках зовнішня шина даних має ту ж ширину, що і АЛП. У 8088 використовувалися 16-бітний АЛП і 8-бітна шина, в той час як сучасні Pentiums обробляють дані по 64 біта за цикл для своїх 32-бітних АЛП.

					БР.171.061.001 ПЗ	Арк.
						18
Змн.	Арк.	№ докум.	Підпис	Дата		

- **Продуктивність (MIPS)** означає «мільйони інструкцій в секунду» і є приблизним показником продуктивності процесора. Сучасні процесори можуть робити багато різних речей, тому рейтинги MIPS втрачають своє значення, в таблиці наведені відносні потужності процесорів.

З таблиці можна бачити зв'язок між тактовою частотою і MIPS. Максимальна тактова частота залежить від виробничого процесу і затримок всередині чіпа. Існує також взаємозв'язок між кількістю транзисторів і MIPS. Наприклад, 8088 працював з тактовою частотою 5 МГц, але виконував операції з частотою 0,33 MIPS (приблизно одна інструкція на 15 тактових циклів). Сучасні процесори часто можуть виконувати операції зі швидкістю дві інструкції за такт. Це поліпшення прямо пов'язане з кількістю транзисторів в чіпі [4].

1.1.2 Значення розміру процесу процесора

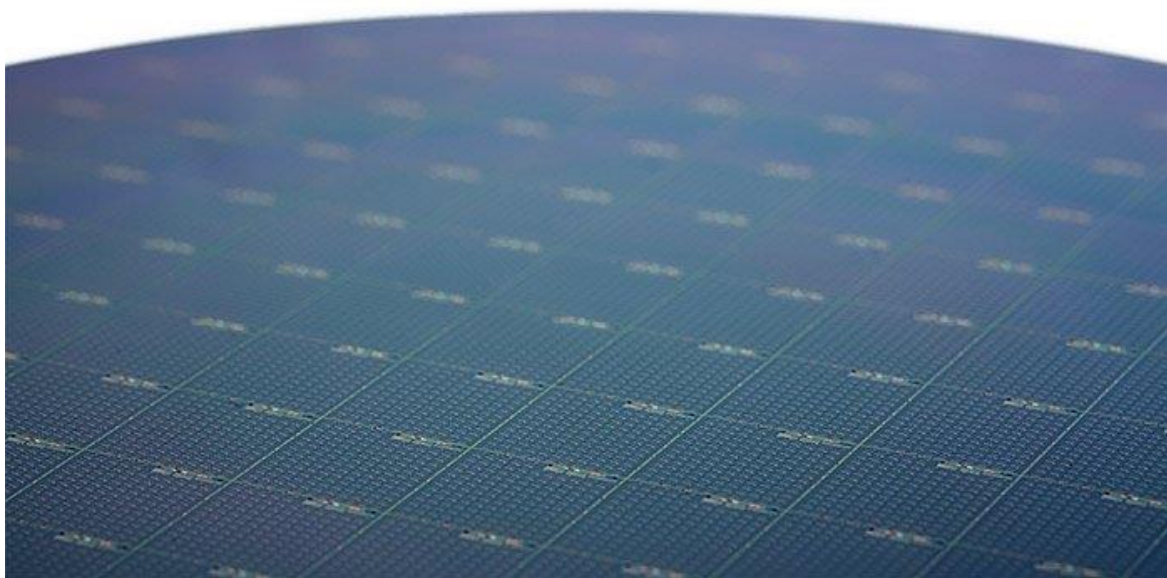


Рисунок 1.2. Зображення процесору Intel.

У цьому контексті «процес» використовується для опису процесу виготовлення, а не процесора комп'ютера. Розглядається виготовлення процесору, а не робота. Розмір вузла процесу, вимірюваний в нанометрах, визначає обсяг найменшого можливого елемента процесора.

Якщо дизайн процесора являє собою цифрове зображення, розмір одного «пікселя» буде розміром процесу. Наприклад, в поточному процесі Intel

					БР.171.061.001 ПЗ	Арк.
						19
Змн.	Арк.	№ докум.	Підпис	Дата		

найменший можливий елемент становить 14 нанометрів (або 14 нм) (рис.1.2). Чим менше процес, тим більше дозвіл, який можна отримати. В результаті виробники можуть зменшити розмір транзисторів і інших компонентів. Це означає, що більша кількість транзисторів може бути вміщено в менший фізичний простір. Це дає деякі основні переваги, а також пару недоліків.

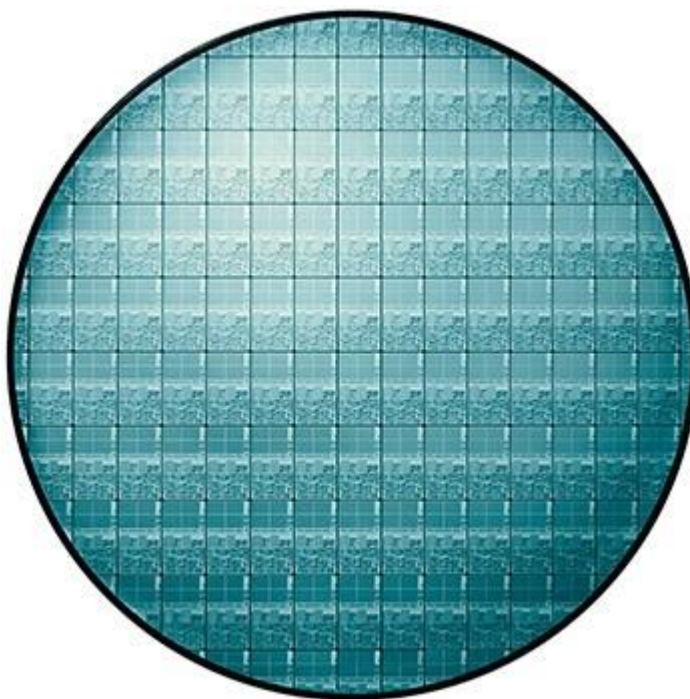


Рисунок 1.3. Чіп виготовлений на круглій кремнієвій пластині.

При стисканні всі частини транзистора однаково, електричні властивості цього транзистора не зміняться. При збільшенні кількості транзисторів вдасться помістити в певному просторі, тим більше можна отримати обчислювальної потужності. Це досягається завдяки збільшенню обчислювального паралелізму і розмірів кеша. Зменшення розміру транзисторів забезпечить можливість прискорити чіп або додати нові функції.

Менші процеси також мають більш низьку ємність, що дозволяє транзисторам швидше вмикатися і вимикатися при меншому споживанні енергії. Чим швидше транзистор може вмикатися і вимикатися, тим швидше він може працювати. Транзистори, які вмикаються і вимикаються з меншим енергоспоживанням, більш ефективні, завдяки зниженню робочої потужності

або «динамічного енергоспоживання», необхідних процесорам. Мікросхема з більш низьким динамічним енергоспоживанням буде розряджати акумулятори повільніше, довше працювати і буде більш екологічною [5].

Менші чіпи також дешевше у виробництві. Наприклад чіпи виготовлені на круглих кремнієвих пластинах, як показано вище (рисунок 1.3). Одна пластина, як правило, буде містити десятки кристалів процесора. Менший розмір процесу дозволить зробити меншим розмір матриці. При меншому розмірі матриці, на одну кремнієву пластину поміститься більше матриць. Це призводить до підвищення ефективності виробництва, зниження витрат на виготовлення. Розробка нового процесу вимагає значних інвестицій, але після того, як ці витрати будуть відшкодовані, витрати на штампи значно знижуються.

Недоліком меншого розміру процесу є збільшення складності технології виготовлення транзисторів меншого розміру. У міру стиснення транзисторів стає все важче створювати чіпи, які працюють з максимально можливою тактовою частотою. Деякі чіпи не зможуть працювати на максимальній швидкості, і ці чіпи будуть «помічені» або позначені як чіпи з більш низькою тактовою частотою або меншим кешом. Менші процеси зазвичай містять більше мікросхем, пов'язаних з більш низькими тактовими частотами, оскільки створення «ідеального» чіпа є більш складним завданням. Виробники намагаються усунути якомога більше проблем, але це часто зводиться до неминучих змін аналогового світу.

Менші транзистори також мають великий «витік». Витік - це вимір того, скільки струму пропускає транзистор в положенні «вимкнено». Це означає, що зі збільшенням витоку збільшується і статичне енергоспоживання, або кількість енергії, яку транзистор споживає в режимі очікування. Чіп з більшим витоком вимагає більше енергії, навіть коли він не активний, розряджає батареї швидше і працює менш ефективно.

У меншого процесу може бути нижчий вихід, що призводить до меншої кількості повністю функціональних чіпів. Це може привести до затримок і дефіциту продукції. Це ускладнює окупність інвестицій, необхідних для

					БР.171.061.001 ПЗ	Арк.
						21
Змн.	Арк.	№ докум.	Підпис	Дата		

розробки нового процесу. Цей елемент ризику лежить в основі будь-якого нового виробничого процесу, але особливо актуальним для такого точного процесу, як виготовлення напівпровідників.

Виробники намагаються зменшити або усунути ці проблеми при розробці нового процесу. Це дає змогу отримувати чіпи, які працюють швидше і ефективніше навіть при зменшенні розміру процесу.

1.1.3 Принцип роботи МП

Щоб зрозуміти, як працює мікропроцесор, потрібно дізнатися про логіку, використану для його створення. В процесі роботи з процесором потрібно ознайомитися з мовою Асемблер - рідна мова мікропроцесора, а також з методами підвищення швидкості процесора.

Мікропроцесор виконує набір машинних інструкцій, які повідомляють процесору, що робити. На підставі інструкцій мікропроцесор виконує три основні дії:

- Використовуючи свій АЛП (Арифметичний / Логічний блок), мікропроцесор може виконувати математичні операції, такі як додавання, віднімання, множення і ділення. Сучасні мікропроцесори містять повні процесори з плаваючою комою, які можуть виконувати надзвичайно складні операції з великими числами з плаваючою комою.
- Процесор може переміщати дані з однієї комірки пам'яті в іншу.
- Процесор може приймати рішення і переходити до нового набору інструкцій, заснованих на цих рішеннях.

					БР.171.061.001 ПЗ	Арк.
						22
Змн.	Арк.	№ докум.	Підпис	Дата		

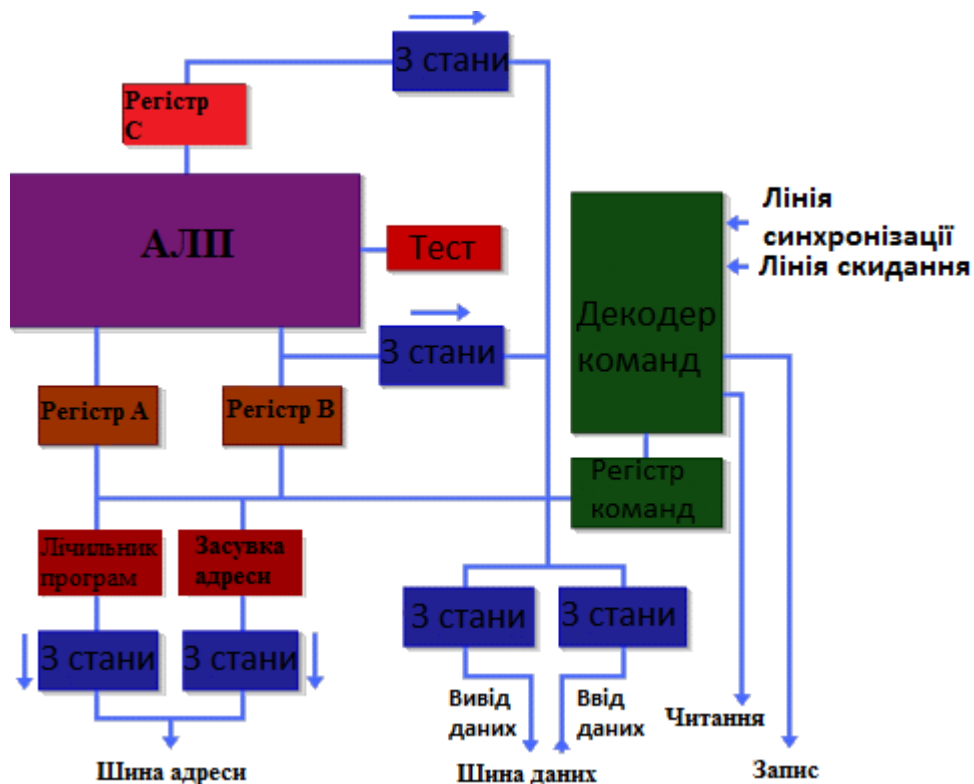


Рисунок 1.4. Структурна схема найпростішого мікропроцесора.

Також існують складні операції, які робить мікропроцесор, але це три його основні дії. Наступна діаграма (рисунок 1.4) показує надзвичайно простий мікропроцесор, здатний виконувати ці три речі.

Даний мікропроцесор має такі складові:

- **Шина адреси** (розміром 8, 16 або 32 біта), яка відправляє адреси в пам'ять;
- **Шина даних** (розміром 8, 16 або 32 біта), яка може відправляти дані в пам'ять або отримувати дані з пам'яті;
- **Рядок RD** (читання) і **WR** (запис), щоб повідомити пам'яті, чи хоче вона встановити або отримати адресу розташування;
- **Лінія синхронізації**, що дозволяє синхронізувати тактовий імпульс процесора;
- **Лінія скидання**, яка скидає лічильник програми на нуль (або щось ще) і відновлює виконання

Припустимо, що шина адреси і даних в цьому прикладі має розмір 8 біт.

Такий процесор складатиметься з даних компонентів:

- **Регістри А, В і С** - це засувки, зроблені з тригерів.
- **Засувка адреси** схожа на регістри А, В і С.
- **Лічильник програм** являє собою засувку з додатковою здатністю збільшуватися на 1 при отриманні відповідної вказівки, а також скидатися в нуль при отриманні відповідної вказівки.
- **АЛП** може бути таким же простим, як 8-бітний, або він може додавати, віднімати, множити і ділити 8-бітні значення.
- **Тестовий регістр** - це спеціальний фіксатор, який може містити значення з порівнянь, виконаних в АЛП. АЛП зазвичай може порівнювати два числа і визначати, чи рівні вони, чи одне більше іншого і т.д. Тестовий регістр також зазвичай може містити біт перенесення з останнього ступеня суматора. Він зберігає ці значення в тригерах, і потім декодер команд може використовувати ці значення для прийняття рішень.
- На діаграмі є шість полів з позначкою «3 стани». Це три стани буферів. Буфер із трьома станами може пропускати 1, 0 або він може по суті відключати свій вихід (уявіть собі перемикач, який повністю відключає вихідну лінію від проводу, до якого прямує вихід). Буфер із трьома станами дозволяє декільком виходам з'єднуватися з проводом, але тільки один з них фактично виводить 1 або 0 на лінію.
- **Регістр команд і декодер команд** відповідають за управління всіма іншими компонентами. Наприклад:
 - Задати регістру А, зафіксувати поточне значення на шині даних.
 - Задати регістру В зафіксувати значення, яке в даний момент знаходиться на шині даних.
 - Задати в регістру С, зафіксувати значення, в даний час виведене АЛП.
 - Задати регістру лічильника програми зафіксувати поточне значення на шині даних.

					БР.171.061.001 ПЗ	Арк.
						24
Змн.	Арк.	№ докум.	Підпис	Дата		

- Задати регістру адреси, зафіксувати значення, яке в даний момент знаходиться на шині даних.
- Скажіть регістру інструкцій, щоб зафіксувати значення, що знаходиться в даний момент на шині даних.
- Задати лічильнику програми збільшити значення
- Задати лічильнику програми обнулитися
- Почати процес, будь-який, з шести буферів з трьома станами (шість окремих рядків).
- Задати АЛП, яку операцію виконати
- Задати контрольному регістру зафіксувати тестові біти АЛП
- Активувати лінію RD
- Активувати лінію WR

У декодер команд входять біти з тестового регістра і лінії синхронізації, а також біти з регістра команд.

1.2 Мікроконтролери

Мікроконтролер (МК) - це автономна система з периферійними пристроями, пам'яттю і процесором, яку можна використовувати в якості вбудованої системи. Більшість програмованих мікроконтролерів, які використовуються сьогодні, вбудовані в інші споживчі товари або механізми, включаючи телефони, периферійні пристрої, автомобілі, побутову техніку та комп'ютерні системи. У зв'язку з цим інша назва мікроконтролера - «вбудований контроллер». Деякі вбудовані системи є більш складними, в той час як інші пред'являють мінімальні вимоги до пам'яті і мають низьку складність програмного забезпечення. Пристрої введення і виведення включають в себе соленоїди, РК-дисплеї, реле, перемикачі та датчики для даних, таких як вологість, температура або рівень освітленості, серед інших [6].

Мікроконтролери є невід'ємною частиною вбудованих систем. Мікроконтролер - це, в основному, дешевий і невеликий комп'ютер на одній мікросхемі, який містить процесор, невелику пам'ять і програмовану периферію

					БР.171.061.001 ПЗ	Арк.
						25
Змн.	Арк.	№ докум.	Підпис	Дата		

введення-виведення. Вони призначені для використання в автоматично контрольованих пристроях і пристроях для виконання попередньо визначених і заздалегідь запрограмованих завдань. Щоб отримати краще уявлення про те, що насправді є мікроконтролером давайте подивимося на приклад продукту, в якому використовується мікроконтролер. Цифровий термометр, який відображає температуру навколишнього середовища, використовує мікроконтролер, який підключений до датчика температури і блоку відображення (наприклад, РК-дисплей). Мікроконтролер тут отримує дані від датчика температури в необробленому вигляді, обробляє їх і відображає на невеликому РК-дисплеї в зрозумілій формі. Один або кілька мікроконтролерів використовуються в багатьох електронних пристроях в залежності від вимог і складності додатків.

На ринку представлений широкий асортимент мікроконтролерів. Різні компанії, такі як Atmel, ARM, Microchip, Texas Instruments, Renesas, Freescale, NXP Semiconductors і ін., виготовляють мікроконтролери різного типу з різними функціями. Вивчаючи різні параметри, такі як програмована пам'ять, напруга живлення, контакти вводу / виводу, швидкість і т.д., можна вибрати відповідний мікроконтролер для їх застосування.

Розглянемо ці параметри:

Шина даних (бітовий розмір):

При класифікації відповідно до розміру бітів більшість мікроконтролерів мають діапазон від 8 до 32 біт (також доступні мікроконтролери з більш високими бітами). У 8-бітному мікроконтролері його шина даних складається з 8 рядків даних, в той час як в 16-бітному мікроконтролері його шина даних складається з 16 рядків даних і т.д. для 32-бітних і більш

Обсяг пам'яті:

Мікроконтролери потребують наявності пам'яті (RAM, ROM, EPROM, EEPROM, флеш-пам'ять і т.д.) для зберігання програм і даних. У той час як деякі мікроконтролери мають вбудовані мікросхеми пам'яті, інші вимагають підключення зовнішньої пам'яті. Їх називають мікроконтролерами з вбудованою пам'яттю і мікроконтролерами із зовнішньою пам'яттю відповідно. Розмір

					БР.171.061.001 ПЗ	Арк.
						26
Змн.	Арк.	№ докум.	Підпис	Дата		

вбудованої пам'яті також різняться в різних типах мікроконтролерів, і, як правило, ви знайдете мікроконтролери з об'ємом пам'яті від 4кБ до 4 МБ.

Кількість вхідних / вихідних контактів:

Мікроконтролери розрізняються залежно від кількості розмірів висновків введення-виведення. Можна вибрати певний мікроконтролер відповідно до вимоги програми [7].

Набір інструкцій:

Існує два типи наборів команд - RISC і CISC. Мікроконтролер може використовувати комп'ютер зі скороченим набором інструкцій (RISC) або комп'ютер з комплексним набором інструкцій (CISC) архітектуру. Як впливає з назви, RISC скорочує час операції, що визначає тактовий цикл інструкції; в той час як CISC дозволяє застосовувати одну інструкцію в якості альтернативи багатьом інструкцій.

Існує два типи мікроконтролерів - мікроконтролери з архітектурою пам'яті Harvard (Гарвардська архітектура - архітектура, головною відмінністю якої є те, що дані та оператори (алгоритм) зберігаються окремо) і мікроконтролери з архітектурою пам'яті Princeton (архітектура фон Неймана - широко відомий принцип спільного зберігання команд і даних в пам'яті комп'ютера).

Ось деякі популярні мікроконтролери серед студентів і любителів:

- Мікроконтролери серії 8051 (8 біт)
- Мікроконтролери AVR від Atmel (серія ATtiny, ATmega)
- Мікроконтролери серії PIC від Microchip
- Мікроконтролери Texas Instruments, такі як MSP430
- ARM Мікроконтролери

1.2.1 Типи мікроконтролерів

Існує кілька різних типів програмованих мікроконтролерів, які класифікуються за кількома параметрами, включаючи біти, розмір флеш-пам'яті, обсяг оперативно запам'ятовуючого пристрою (ОЗП), кількість рядків введення / виведення, тип упаковки, напруга живлення і швидкість роботи.

					БР.171.061.001 ПЗ	Арк.
						27
Змн.	Арк.	№ докум.	Підпис	Дата		

Програмовані мікроконтролери містять порти введення / виведення загального призначення. Кількість цих портів варіюється в залежності від мікроконтролера. Вони можуть бути налаштовані на вхід або вихід за допомогою програмного забезпечення. Якщо вони налаштовані на вхідний стан, ці контакти можуть використовуватися для зчитування зовнішніх сигналів або датчиків. Якщо вони налаштовані на вихідний стан, вони можуть управляти зовнішніми пристроями, такими як світлодіодні дисплеї і двигуни.

Програмовані мікроконтролери призначені для використання у вбудованих пристроях, на відміну від мікропроцесорів, які можна знайти в ПК. Мікроконтролери використовуються в пристроях з автоматичним управлінням, включаючи електроінструменти, іграшки, вбудовані в медичні пристрої, офісні машини, системи управління двигуном, побутові прилади, пульти дистанційного керування та інші типи вбудованих систем.

Розглянемо основні блоки що беруть участь в роботі мікроконтролера:

- ПЗП програми і ПЗП даних

Вбудована пам'ять ПЗП (Read Only Memory) на мікроконтролері нагадує жорсткий диск мікроконтролера. У нього є два розділи. Один розділ зарезервований для зберігання програмного коду, а інший розділ зарезервований для постійного зберігання даних, які використовуються мікросхемою під час нормального виконання програми.

- ЦП

ЦП позначає центральний процесор. Це в основному «мізки» мікроконтролера. Це те, що вибирає інструкції з пам'яті коду і виконує ці інструкції.

- ОЗУ даних

ОЗУ даних (оперативний запам'ятовуючий пристрій) - це простір даних, який використовується для тимчасового зберігання констант і значень змінних, які використовуються мікроконтролером під час нормального виконання програми. Обсяг фізичного ОЗУ в мікроконтролері. ОЗУ даних на мікроконтролері організовано в декілька «регістрів», кожен зі своєю унікальною

					БР.171.061.001 ПЗ	Арк.
						28
Змн.	Арк.	№ докум.	Підпис	Дата		

«адресою». Регістр ОЗУ на 8-бітному мікроконтролері може містити всього 8 біт або один байт даних. Типова специфікація простору ОЗУ може вказувати, що вона становить 256×8 . Це означає, що в ОЗУ є в цілому 256 регістрів, і ці регістри можуть містити по 8 бітів кожен.

- Регістр - це просто місце в пам'яті, в яке ви можете записувати або зчитувати дані. Деякі з нас називають регістри «місцями розташування».

1.2.2 Регістри спеціальних функцій

Регістри спеціальних функцій (SFR) на мікроконтролері аналогічні регістрам в оперативній пам'яті даних. Є можливість запису дані в них, а також читати дані з них. Відмінності між ними полягають в тому, що деякі SFR безпосередньо керують вбудованим обладнанням мікроконтролера, в той час як інші контролюються вбудованим обладнанням мікроконтролера.

Кожен біт в SFR призначається функції. У SFR є контрольні біти і біти прапора. Керуючі біти схожі на «перемикачі», які вмикають або вимикають функцію в залежності від того, записана 1 або 0 в цю позицію біта в SFR. Біти прапора схожі на «світлові індикатори», які вказують, чи існує ця умова в залежності від того, чи є біт прапора 1 або 0. Біти управління безпосередньо керують апаратним забезпеченням. Прапорні біти контролюються апаратними засобами. У будь-якій конкретній програмі зазвичай записуються керуючі біти, поки читаються біти прапора (деякі біти прапора повинні бути очищені вручну, шляхом запису в них в залежності від) [8].

Кожному елементу апаратного забезпечення на мікроконтролері буде присвоєно як мінімум 1 SFR. Деякому обладнанню може бути призначено кілька SFR.

1.3 Відмінності між мікропроцесором і мікроконтролером

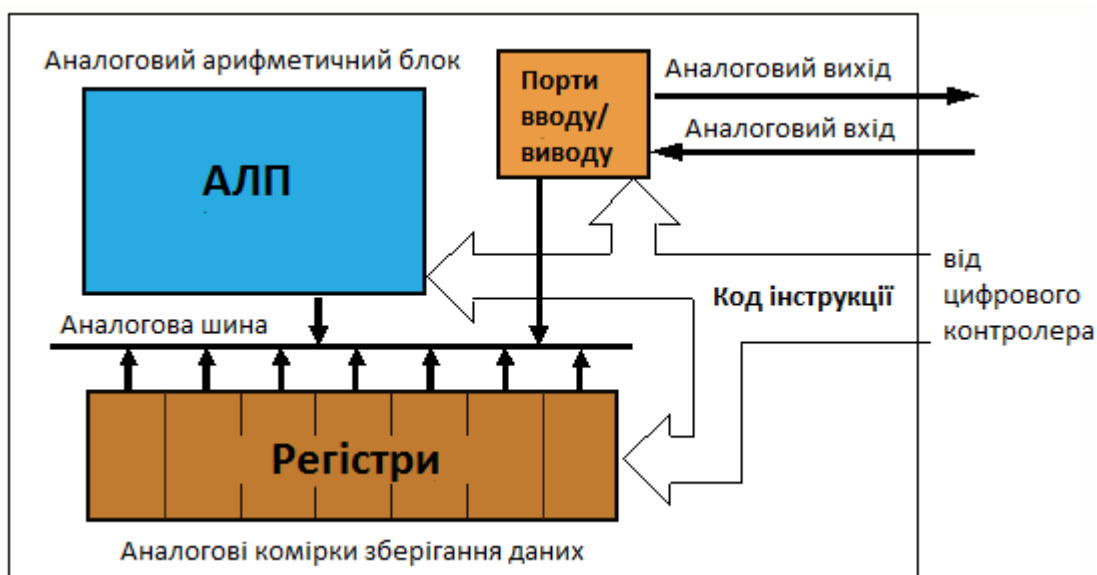


Рисунок 1.5. Структурна схема мікропроцесора.

Мікропроцесор - це мікросхема, в якій є тільки процесор, тобто тільки обчислювальні потужності, такі як Intel Pentium 1,2,3,4, Core 2 Duo, I3, I5 і т.д. Ці мікропроцесори не мають оперативної пам'яті, ПЗУ і інших периферійних пристроїв на чіпі (рисунок 1.5). Розробник системи повинен додати їх зовні, щоб зробити їх більш функціональними.

Але це не відноситься до мікроконтролерів. Мікроконтролер має процесор, а також фіксований обсяг оперативної пам'яті, ПЗУ та інших периферійних пристроїв, вбудованих в один чіп. Час від часу це також називають міні-комп'ютером або комп'ютером на одній мікросхемі. Сьогодні різні виробники випускають мікроконтролери з широким спектром функцій, доступних в різних версіях [9].

Найвідоміші виробники: ATMEL, Microchip, TI, Freescale, Philips, Motorola і т.д.

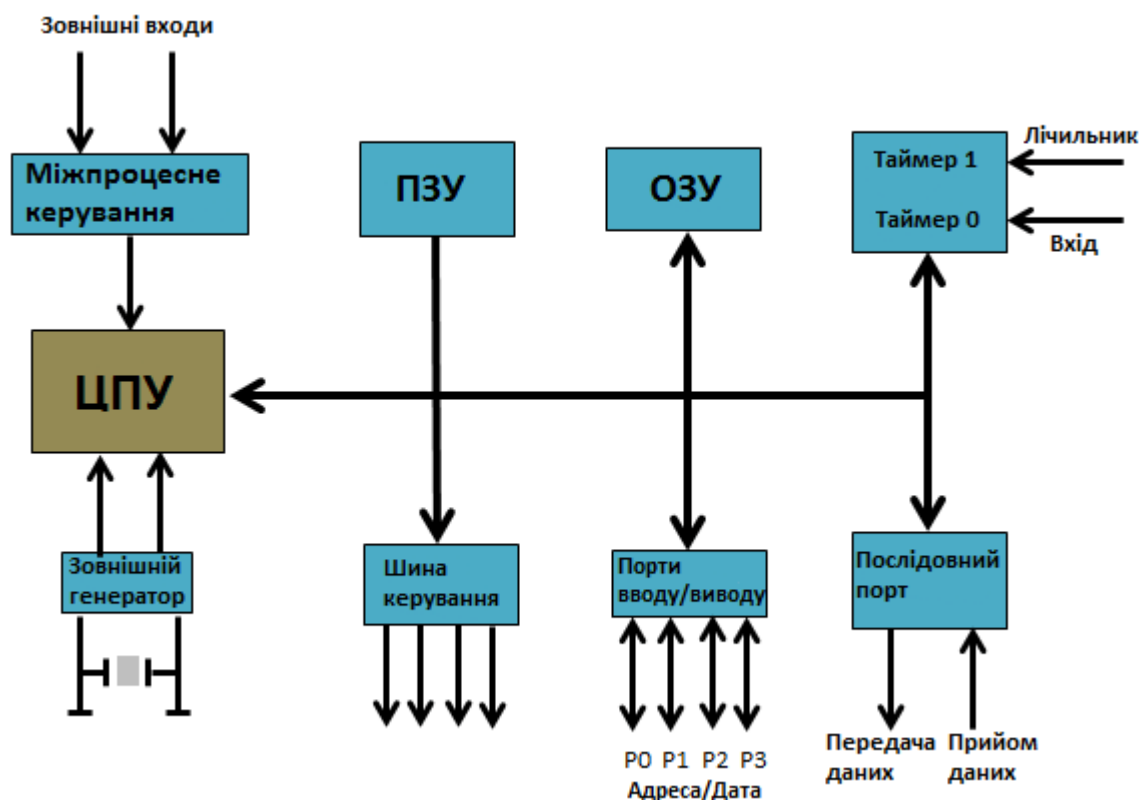


Рисунок 1.6. Структурна схема мікропроконтролера.

Мікроконтролери призначені для виконання конкретних завдань. Конкретні засоби застосування, де визначено взаємозв'язок введення і виведення. Залежно від введення, деяка обробка повинна бути зроблена, і висновок доставлений. Наприклад, клавіатура, миша, пральна машина, цифрова камера, пульт дистанційного керування, мікрохвильова піч, автомобілі, велосипеди, телефон, мобільні телефони, годинники і т.д. Оскільки додатки дуже специфічні, їм потрібні невеликі ресурси, такі як ОЗУ, ПЗП, порти вводу / виводу і т.д. і, отже, можуть бути вбудовані в один чіп (рисунок 1.6). Це, в свою чергу, зменшує розмір і вартість.

Мікропроцесори знаходять застосування, де завдання не є специфічними, такі як розробка програмного забезпечення, ігор, веб-сайтів, редагування фотографій, створення документів і т.д. В таких випадках взаємозв'язок між вводом і виводом не визначається. Їм потрібна велика кількість ресурсів, таких як ОЗП, ПЗП, порти вводу / виводу і т.д.

Тактова частота процесора досить висока в порівнянні з мікроконтролером. У той час як мікроконтролери працюють від декількох МГц до 30-50 МГц, сучасні мікропроцесори працюють на частотах вище 1 ГГц, виконуючи складні завдання.

Мікроконтролер набагато дешевший, ніж мікропроцесор. Однак замість мікропроцесора можна використовувати мікроконтролер, а замість мікроконтролера не рекомендується використовувати мікропроцесор, оскільки це робить додаток досить дорогим. Мікропроцесор не можна використовувати окремо, оскільки всі периферійні пристрої мікроконтролера знаходяться на одному кристалі, він компактний, а мікропроцесор громіздкий [10].

Мікроконтролери виготовляються з використанням комплементарної металооксидних напівпровідникової технології, тому вони набагато дешевше мікропроцесорів. Крім того, додатки, створені за допомогою мікроконтролерів, дешевше, оскільки їм потрібно менше зовнішніх компонентів, в той час як загальна вартість систем, виготовлених з мікропроцесорами, висока через велику кількість зовнішніх компонентів, необхідних для таких систем.

Як правило, мікроконтролери мають систему енергозбереження, таку як режим очікування або режим економного споживання, тому в цілому він споживає менше енергії, а також через низький рівень зовнішніх компонентів загальне споживання енергії менше. У той час як в мікропроцесорах, як правило, немає системи енергозбереження, а також використовується багато зовнішніх компонентів, тому її енергоспоживання є високим у порівнянні з мікроконтролерами.

Мікроконтролери компактні, що робить їх вигідною і ефективною системою для невеликих продуктів і додатків, в той час як мікропроцесори громіздкі, тому вони краще для великих додатків.

Конструкція мікроконтролера залежить від архітектури Гарварда, де пам'ять для програм та інформації специфічна, в той час як мікропроцесори залежать від архітектури фон Неймана, в якій обидві пам'яті зберігаються в еквівалентному модулі пам'яті.

					БР.171.061.001 ПЗ	Арк.
						32
Змн.	Арк.	№ докум.	Підпис	Дата		

1.4 ATmega32 - 8-бітний мікроконтролер AVR



Рисунок 1.7. Мікроконтролер ATМega32А.

Мікроконтролери AVR засновані на сучасній архітектурі RISC. ATmega32 - 8-бітний CMOS-мікроконтролер з низьким енергоспоживанням, заснований на розширеній архітектурі RISC AVR (рисунок 1.7). AVR може виконувати 1 мільйон інструкцій в секунду, якщо частота циклу становить 1 МГц.

Ключова особливість:

- 32 x 8 регістрів загального призначення.
- 32 Кбайт в системі самопрограмуємої флеш-пам'яті програм
- 2 Кбайт внутрішньої SRAM
- 1024 байти EEPROM
- Можна користуватися в 40-контактному DIP, 44-контактному QFP, 44-контактному QFN / MLF
- 32 програмовані лінії введення / виводу
- 8-канальний, 10-бітний АЦП
- Два 8-бітних таймера / лічильника з окремими прескалерами і режимами порівняння

- Один 16-розрядний таймер / лічильник з окремим прескалером, режимом порівняння і режимом захоплення.

- 4 канали ШІМ

- У системному програмуванні за допомогою вбудованої програми завантаження

- Програмований сторожовий таймер з окремим вбудованим генератором.

- Програмований серійний USART

- Послідовний інтерфейс ведучого / підлеглого SPI

Особливості мікроконтролера:

- Шість сплячих режимів: режим очікування, шумозаглушення АЦП, енергозбереження, відключення живлення, режим очікування і розширений режим очікування.

- Внутрішній калібрований RC генератор

- Зовнішні та внутрішні джерела переривань

- Скидання при включенні живлення і програмованому відключенні.

Всі 32 регістри безпосередньо підключені до арифметичного логічного блоку (АЛП), що дозволяє отримати доступ до двох незалежних регістрів в одній інструкції, що виконується за один такт.

Відключення живлення зберігає вміст регістру, але зупиняє генератор. Всі інші функції мікросхеми будуть відключені до наступного зовнішнього переривання. Асинхронний таймер дозволяє користувачеві підтримувати таймер на основі режиму енергозбереження, поки інша частина пристрою знаходиться в сплячому режимі [11].

Режим шумозаглушення АЦП зупиняє ЦП і всі модулі введення / виводу, крім АЦП і асинхронного таймера. У режимі очікування, крім кварцового генератора, інша частина пристрою знаходиться в сплячому режимі. Проте основний генератор, і асинхронний таймер продовжують працювати в розширеному режимі очікування.

					БР.171.061.001 ПЗ	Арк.
						34
Змн.	Арк.	№ докум.	Підпис	Дата		

АТmega32 є потужним мікроконтролером завдяки вбудованій самопрограмований флеш-пам'яті на монолітній мікросхемі, яка забезпечує гнучке і економічно ефективне рішення для багатьох вбудованих додатків управління.

У системної програмованої флеш-пам'яті програм:

АТmega32 містить 32 Кбайт на кристалі в пам'яті, з можливістю перепрограмування, системи для зберігання програм. Flash організований як 16k X 16, і його пам'ять розділена на два розділи: розділ Boot-програми та розділ програми-програми.

1.5 Огляд існуючих прототипів

Мікропроцесорна система (МПС) на основі мікроконтролера (МК) використовуються найчастіше як вбудованих систем для вирішення завдань управління деяким об'єктом. Важливою особливістю даного застосування є робота в реальному часі, тобто забезпечення реакції на зовнішні події протягом визначеного часового інтервалу. Такі пристрої отримали назву контролерів.

Перед розробником МПС стоїть завдання реалізації повного циклу проектування, починаючи від розробки алгоритму функціонування і закінчуючи комплексними випробуваннями в складі виробу.

На просторах Інтернету можна зустріти безмежну кількість варіантів налагоджувальних плат. Серед них є плати дуже складні, які мають в своєму складі велику кількість елементів та складну конструкцію. Зразок складної налагоджувальної плати представлено на рисунку 1.8. Також в мережі Інтернет можна знайти і прості приклади плат, проте в їх складі немає всіх необхідних для відмінного функціонування компонентів.

					БР.171.061.001 ПЗ	Арк.
						35
Змн.	Арк.	№ докум.	Підпис	Дата		

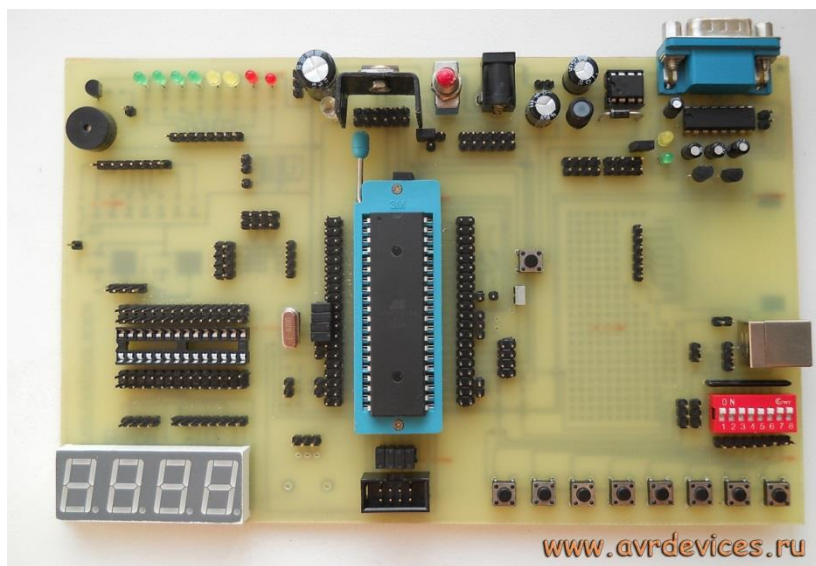


Рисунок 1.8. Складна налагоджувальна плата.

При розробці власної налагоджувальної плати необхідно виходити з того, що максимальне використання апаратних засобів спрощує розробку і забезпечує високу швидкодію контролера в цілому, але супроводжується, як правило, збільшенням вартості і споживаної потужності.

При виборі типу МК враховуються такі основні характеристики:

- розрядність;
- швидкодія;
- набір команд і способів адресації;
- вимоги до джерела живлення і споживана потужність в різних режимах;
- обсяг ПЗП програм і ОЗП даних;
- можливості розширення пам'яті програм і даних;
- наявність і можливості периферійних пристроїв, включаючи засоби підтримки роботи в реальному часі (таймери, процесори подій і т.п.);
- можливість перепрограмування в складі пристрою;
- наявність і надійність засобів захисту внутрішньої інформації;
- вартість;

- наявність і доступність ефективних засобів програмування і налагодження МК;
- доступність можливості заміни виробами інших фірм.

Найбільш відомими брендами в області виробництва налагоджувальних плат і модулів розширення є наступні компанії: Atmel, Digilent, Embest, Freescale, Future Technology, IAR Systems, Microchip, mikroElektronika, Olimex, Seeed Studio, ST Microelectronics, Texas Instruments, Waveshare.

Сучасна стратегія модульного проектування забезпечує споживача різноманітністю моделей МК з одним і тим же процесорним ядром. Така структурна і різноманітність відкриває перед розробником можливість вибору оптимального МК.

Світові фірми пропонують широкий вибір налагоджувальних плат, які мають більше можливостей, але вартість таких плат є дуже високою. Зразок плати виробника Atmel зображений на рисунку 1.9.

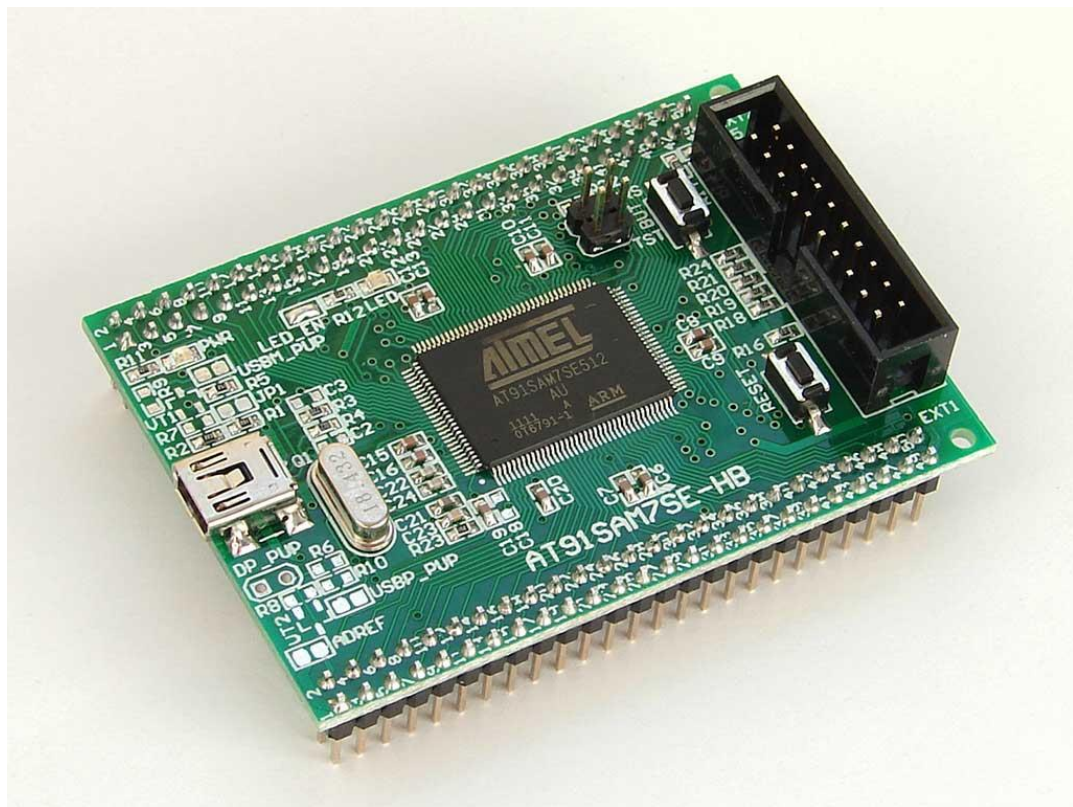


Рисунок 1.9. Налагоджувальна плата виробника Atmel [12].

Компанія Atmel у співпраці з розробниками команди Arduino на конференції BUILD анонсувала кілька налагоджувальних платформ на базі мікроконтролерів AVR (ATmega32U4 і AVR UC3) і мікроконтролера з ядром Cortex-M3 SAM3U.

					БР.171.061.001 ПЗ	Арк.
						38
Змн.	Арк.	№ докум.	Підпис	Дата		

Висновки до розділу 1

При огляді науково-технічної літератури було проаналізовано принцип роботи мікропроцесора та мікроконтролера, відмінності між мікропроцесором і мікроконтролером, різновиди мікропроцесорів компанії Intel, типи мікроконтролерів.

Особливу увагу приділено: значенню розмірів процесу процесора, складовим компонентам мікропроцесорів, значенню регістрів спеціальних функцій.

Проаналізовано основні характеристики та переваги мікроконтролера ATmega32A, який використовувався як основа електронного стенду при виконанні дипломного проекту

Зроблено огляд існуючих прототипів електронних стендів для налагодження роботи та візуальної перевірки роботи програм мікроконтролерів.

					БР.171.061.001 ПЗ	Арк.
						39
Змн.	Арк.	№ докум.	Підпис	Дата		

2. РОЗРОБКА ТА ОПИС СТЕНДУ

2.1 Обґрунтування схемних рішень та використаної елементної бази

Повна принципова схема електронного стенду (ЕС) складається після розробки принципових схем функціональних елементів (блоків) структурної схеми. Синтез схеми відбувається в процесі всебічного аналізу властивостей і параметрів описаних в літературі схем, з'ясування їх відповідності вимогам, що пред'являються, вибору найбільш відповідної цим вимогам схеми.

Розробка пристрою буде проводитися з урахуванням таких вимог:

- простота схеми (мінімальна кількість компонентів);
- функціональна насиченість, різноманіття регульованих параметрів;
- стійкість до змін напруги, довговічність;
- можливість керування мінімум чотирма функціями;
- низьке енергоспоживання.
- При розробці конструкції виробу необхідно:
- спроектувати схеми лабораторного стенду на основі мікроконтролера АТМega32А;
- вибрати радіоелементи, що забезпечують надійну і стійку роботу схеми;
- прагнути до скорочення номенклатури деталей і елементів, тобто зробити конструкцію максимально легкою і компактною;
- мати оптимальне співвідношення ціни та якості вибраних електро-радіо елементів (ЕРЕ);
- ретельно проаналізувати вимоги до приладу, приступаючи до розробки конструкції.

Усі використані ЕРЕ є стандартними, тобто вони є покупними, що значно збільшує надійність виробу.

При розробці друкованої плати враховуємо ширину друкованих провідників, отворів для свердління та розміри кожного елемента, щоб відповідати правилам оптимальної конфігурації [13].

					БР.171.061.001 ПЗ	Арк.
						40
Змн.	Арк.	№ докум.	Підпис	Дата		

2.2 Опис і робота стенду

2.2.1 Призначення стенду

При всій різниці у найменуваннях, всі налагоджувальні плати виконують одну функцію: зменшують ймовірність непрацездатності програми через неправильне функціонування апаратної частини проекту – «заліза». Розробник як би відсувається від апаратної частини проекту, і концентрує свої зусилля на процесі написання і налагодження програми [14].

При вивченні програмування мікроконтролерів дуже важливим є поєднання теорії з практикою, реалізація кожної програми на практиці. Це можна здійснити різними шляхами, наприклад, тестувати на симуляторах чи збирати на макеті. Але в цих двох методів є суттєві недоліки: робота в симуляторі часто відрізняється від роботи з залізом, при збиранні на макетній платі можна зробити помилку і вивести контролер з ладу або підключити невірно [15].

Відмінним рішенням для усунення цих проблем є налагоджувальна плата. Це плата, де вже все правильно підключено і розведено. Функціонування заліза можна перевірити на еталонній демопрограмі, а вбудовані системи діагностики і прошивки дозволяють заливати нову прошивку в кристал одним рухом. Залишається тільки експериментувати, не відволікаючись на сторонні фактори.

Також налагоджувальна плата служить зручним засобом для швидкого прототипування та обкатки вузлів, перевірки ідей і методів. Зібрати на демоплаті, перевірити, що ідея працює, як потрібно підкоригувати, а після вже налагоджений код використати в реальному проекті.

Електронний стенд призначений для наглядної перевірки програм. Він дозволяє проводити демонстрацію різних програм, наприклад: бігучий вогник, запалювання світлодіодів, таймер, світлофор. І цей список можна продовжувати далі. Це все можливо завдяки використанню мікроконтролера ATmega32A, що надає великі можливості для програмування. Також в схемі передбачено підключення додаткових периферійних пристроїв, що також сильно збільшує діапазон можливих програм для демонстрації, але це не всі його можливості,

					БР.171.061.001 ПЗ	Арк.
						41
Змн.	Арк.	№ докум.	Підпис	Дата		

завдяки встановленню гнізда, тип мікроконтролера можна легко замінити якщо він має такий самий корпус, що значно розширює можливості стенду.

Стенд що розробляється в даному дипломному проекті може стати базою для створення лабораторних практикумів для поглиблення теоретичних знань студентів при вивченні мікроконтролерів та мікроконтролерних системи та перевірки цих знань на практиці.

Даний стенд дає можливість ознайомитися з основними характеристиками мікроконтролера сімейства AVR ATMega32A.

Недоліки: у роботі налагоджувальної плати застосовується восьмирозрядний мікроконтролер сімейства AVR. Вони популярні завдяки низькій ціні, наявності докладної документації, простоті програмування і легкості монтажу. Однак досить часто бувають випадки, що потужності такого мікроконтролера для вирішення поставленої задачі недостатньо.

Крім швидкості, восьмирозрядні мікроконтролери мають і інші обмеження, наприклад, в багатьох моделях AVR всього один апаратний послідовний порт, що не дозволяє отримувати інформацію від зовнішнього пристрою і одночасно пересилати результати її обробки споживачеві. Наприклад, виведення інформації на графічний індикатор потребує великих ресурсів як швидкості, так і пам'яті.

Але розроблений виріб призначений для використання у навчальних цілях і вирішення задач, які не потребують для свого виконання великих потужностей.

Переваги:

- зручність виконання робіт, які пов'язані із програмуванням мікроконтролерів сімейства AVR;
- можливість удосконалення;
- низький рівень енергоспоживання;
- можливість виконання в одному комплексі декількох задач;
- низька собівартість.

					БР.171.061.001 ПЗ	Арк.
						42
Змн.	Арк.	№ докум.	Підпис	Дата		

2.2.2 Характеристики стенду

Кожен електротехнічний виріб має певні електричні та конструктивні параметри. [16]

В роботі використовується 8 - розрядний мікроконтролер сімейства AVR ATМega32А. Мікроконтролер виготовлений по КМОП - технології, яка в поєднанні з RICS архітектурою дозволяє досягти найкращого співвідношення показників швидкодія / енергоспоживання.

Основні характеристики мікроконтролера що лежить в основі стенду:

- має 131 потужну інструкцію, більшість з яких виконуються за один машинний цикл;
- має 32×8-розрядних регістрів загального призначення і регістри управління вбудованою периферією;
- повністю статична робота;
- продуктивність до 16 мільйонів в секунду операцій при тактовій частоті 16 МГц;
- має вбудований помножуючий пристрій що виконує множення за 2 машинних цикли;
- має шістнадцять кбайт внутрішньосистемної перепрограмованої флеш-пам'яті;
- пам'ять даних (оперативна пам'ять (ОЗП)) 512 байт;
- пам'ять даних (EEPROM) 512 байт;
- програмований захист коду програми;
- два 8-розрядних таймера-лічильника з роздільними подільниками і режимами порівняння;
- є один розширений 16-разрядний таймер-лічильник з окремим подільниками, режимом порівняння і режимом захоплення;
- є лічильник реального часу з окремим генератором;
- є два 8-разрядних канали широтно-імпульсної модуляції (ШІМ);
- є модулятор виходів порівняння;

					БР.171.061.001 ПЗ	Арк.
						43
Змн.	Арк.	№ докум.	Підпис	Дата		

- є 8 мультиплексованих каналів 10-розрядного аналогово-цифрового перетворення;
- є двопровідний послідовний інтерфейс, орієнтований на передачу даних в байтовому форматі;
- є послідовний інтерфейс SPI з підтримкою режимів ведучий / підлеглий;
- є програмований сторожовий таймер з вбудованим генератором;
- є вбудований аналоговий компаратор;
- скидання при подачі живлення і програмована схема скидання при зниженні напруги живлення;
- є вбудований калібрований RC-генератор;
- є зовнішні та внутрішні джерела переривань;
- є програмний вибір тактової частоти;
- напруга живлення $4,5 \div 5,5$ В;
- струм при частоті 8МГц, при температурі 25 °С і напрузі живлення 5В рівний 13 мА. [17]

Основними характеристиками розробленого виробу є:

- напруга живлення $9 \div 12$ В – підключення через зелену клему, 5 В – підключення за допомогою USB;
- використовується мікроконтролер сімейства AVR ATmega32A;
- передбачене під'єднання периферійного обладнання;
- безпосередньо до МК під'єднані світлодіоди (8 штук), семисегментний індикатор та матриця кнопок 4x4;
- передбачена можливість перепрограмування МК за допомогою SPI інтерфейсу.
- конструктивні характеристики виробу:
- плату виконано з одностороннього склотекстоліту завтовшки 1,5 мм з довжиною 65 мм та шириною 35 мм;
- плату закріплено гвинтами.

					БР.171.061.001 ПЗ	Арк.
						44
Змн.	Арк.	№ докум.	Підпис	Дата		

2.3 Склад стенду, призначення складових частин

В своєму складі лабораторний стенд має таку елементну базу: резистори, конденсатори, діоди, стабілізатор, фотодіоди, мікроконтролер, транзистри, кварц, кнопки, штекерні роз'єми, семисегментний індикатор та USB-штекер, що використовуються як допоміжне джерело живлення. Усі перелічені електрорадіоелементи є стандартними, тобто вони є покупними, що значно збільшує технологічність виробу. Семисегментний індикатор і світлодіоди потрібні для світлової індикації в залежності від потрібної програми. Підключати світлодіод безпосередньо, без резистора, можна тільки до джерела живлення, що має великий внутрішній опір. У всіх інших випадках обмеження струму необхідно для запобігання виходу діода з ладу. Для цього ми ставимо по резистору на кожен вивід індикатора (крім 5 і 10 виводу) і по одному резистору на катод світлодіода. Чотири резистора по 10 кОм, чотири діоди і 16 кнопок формують матрицю кнопок, що в подальшому буде потрібна для задання програми. Транзистори потрібні для створення динамічної індикації, що дозволить не перенавантажувати мікроконтролер. Штекерні виводи потрібні для відключення додаткових периферійних плат.

МК виготовлений з використанням технології незалежної пам'яті високої щільності фірми Atmel. Використана технологія Flash дозволяє пам'ять програми перепрограмувати в системі через послідовний інтерфейс SPI, за допомогою звичайної незалежної пам'яті, або за допомогою завантаження програми на чіп що працює на ядрі AVR.

Програма завантаження може використовувати будь-який інтерфейс для завантаження прикладної програми в Flash-пам'яті. Програмне забезпечення в розділі BootFlash буде продовжувати працювати в той час як розділ застосування флеш оновлюється, забезпечуючи справжню операцію читання. Об'єднуючи 8-розрядний RISC-процесор з системою самообслуговування, програмованою флеш-пам'яттю на монолітному чипі, Atmel ATMega32A являє собою потужний мікроконтролер, який забезпечує високо гнучке і економічно ефективне рішення для багатьох вбудованих додатків управління.

					БР.171.061.001 ПЗ	Арк.
						45
Змн.	Арк.	№ докум.	Підпис	Дата		

Розроблений стенд працює на основі кварцового генератора, який призначений для отримання коливань фіксованої частоти з високою температурною та часовою стабільністю, низьким рівнем фазових шумів. В результаті генератор забезпечує високу точність вимірювань.

2.3.1 Будова та робота

Щоб почати роботу із стендом, потрібно подати напругу від 9В до 12В на клему XS1, на виході стабілізатора DA1 отримуємо 5В для живлення МК та інших елементів плати.

Проте конструкція стенду дозволяє подати напругу +5В від ПК за допомогою USB-подовжувача. Про наявність напруги +5В буде свідчити випромінення світло діода «Живлення» VD1.

Для продовження роботи потрібно ознайомитися із завданням відповідної лабораторної роботи. Після чого, розробляємо та записуємо її за допомогою SPI інтерфейсу. Кнопка RESET надає можливість запустити програму спочатку.

Якщо всі дії були виконані правильно, то на стенді можна спостерігати виконання програми (згідно завдання лабораторної роботи).

Програми можна розробляти для роботи світлодіодів VD2-VD9, які під'єднані до порту А мікроконтролера, індикатора DA2, що під'єднаний до порту С, або матриці кнопок, яка під'єднана до порту D.

Є можливість вивести результати кожної обробки МК на периферійні прилади, що можливо під'єднати за допомогою роз'ємів XP1,XP4, XP5 .

Порти введення/виведення AVR мають число незалежних ліній "вхід / вихід" від 3 до 53. Кожна лінія порту може бути запрограмована на вхід або на вихід. Потужні вихідні драйвери забезпечують струмову навантажувальну здатність 20 мА на лінію порту при максимальному значенні 40 мА, що дозволяє, наприклад, безпосередньо підключати до мікроконтролера світлодіоди і біполярні транзистори. Загальне струмове навантаження на всі лінії одного порту не повинне перевищувати 80 мА.

					БР.171.061.001 ПЗ	Арк.
						46
Змн.	Арк.	№ докум.	Підпис	Дата		

В стенді розташований кварц ZQ1, що дозволяє працювати МК на частоті 8МГц (за замовчуванням МК працює на базі внутрішнього RC-кола з частотою 1МГц). Генератор виробляє імпульси для синхронізації роботи всіх вузлів мікроконтролера. Внутрішній тактовий генератор AVR може запускатися від декількох джерел опорної частоти (зовнішній генератор, зовнішній кварцовий резонатор, внутрішній або зовнішній RC-ланцюги).

Джампери JP1-JP10 дають можливість від'єднання світлодіодів VD2-VD9 від порту А, та індикатора DA2 від порту С, для вільного керування периферійними пристроями під'єднаними до цього порту.

Таким чином, за рахунок джамперів, роз'ємів досягається функціональність, гнучкість та універсальність мікропроцесорної системи.

					БР.171.061.001 ПЗ	Арк.
						47
Змн.	Арк.	№ докум.	Підпис	Дата		

Висновки до розділу 2

1. Розроблений стенд для налагодження програм має переваги у порівнянні з заводським аналогом так як комплектація стенду підбирається під виконання визначених завдань, а в стендах які виготовляються спеціалізованими компаніями комплектація є визначеною і зазвичай зміні чи доопрацюванню не підлягає.

2. Розроблений стенд на основі мікроконтролера для налагодження програм також має такі переваги:

- зручність виконання лабораторних робіт, які пов'язані із програмуванням мікроконтролерів сімейства AVR;
- можливість удосконалення;
- низький рівень енергоспоживання;
- можливість виконання в одному комплексі декількох задач;
- низька собівартість.

3. Наведений детальний опис будови стенду, роботи функціональних частин та призначення окремих елементів.

					БР.171.061.001 ПЗ	Арк.
						48
Змн.	Арк.	№ докум.	Підпис	Дата		

3 РОЗРОБКА КОНСТРУКТОРСЬКО-ТЕХНІЧНОЇ ДОКУМЕНТАЦІЇ

3.1 Програмні середовища

Для розробки конструкторсько-технічної документації застосовуються всі можливі різновиди програмного забезпечення, що дозволяє працювати (створювати/редагувати) файли різного формату, відповідно до типу документу що потрібно створити. Для цього завдання ідеально підходять стандартні набори програм, такі як: Microsoft Word, Splan70, AVR Studio 4, layout50. Також для розробки схем та технічних креслень підходить більш спеціалізоване програмне забезпечення, але воно є більш складним, що пояснюється більшим набором функцій, а також воно зазвичай ліцензоване, тобто для використання обов'язково потрібно мати ліцензійний ключ. Кожна програма дозволяє розробляти окремий тип документації, наприклад в Microsoft Word неможливо створити принципову схему чи креслення друкованої плати, для виконання такого завдання слід використовувати програми Splan70 та layout50 відповідно. В наступному пункті розглянемо детальніше процес розробки креслень, використані програмні середовища, та безпосередньо процес виготовлення стенду

3.2. Розробка складального креслення та плати друкованої електронного стенду на основі мікроконтролера для налагодження програм.

Відповідно до завдання, в додатках дипломного проекту повинна міститися така конструкторська документація:

- 1) схема електрична принципова з переліком елементів;
- 2) креслення друкованої плати.

При оформленні графічної частини необхідно дотримуємось основних положень Єдиної системи конструкторської документації (ЄСКД).

Схема електрична принципова – графічний конструкторський документ, який відображає всі основні частини виробу електротехнічного призначення і зв'язки між ними, дає детальне уявлення про принципи роботи виробу.

					БР.171.061.001 ПЗ	Арк.
						49
Змн.	Арк.	№ докум.	Підпис	Дата		

Для визначення складу елементів схеми електричної принципової і відомостей про них, на окремих аркушах формату А4 виконується перелік елементів, який оформляється над основним написом у вигляді таблиці, заповненої зверху вниз.

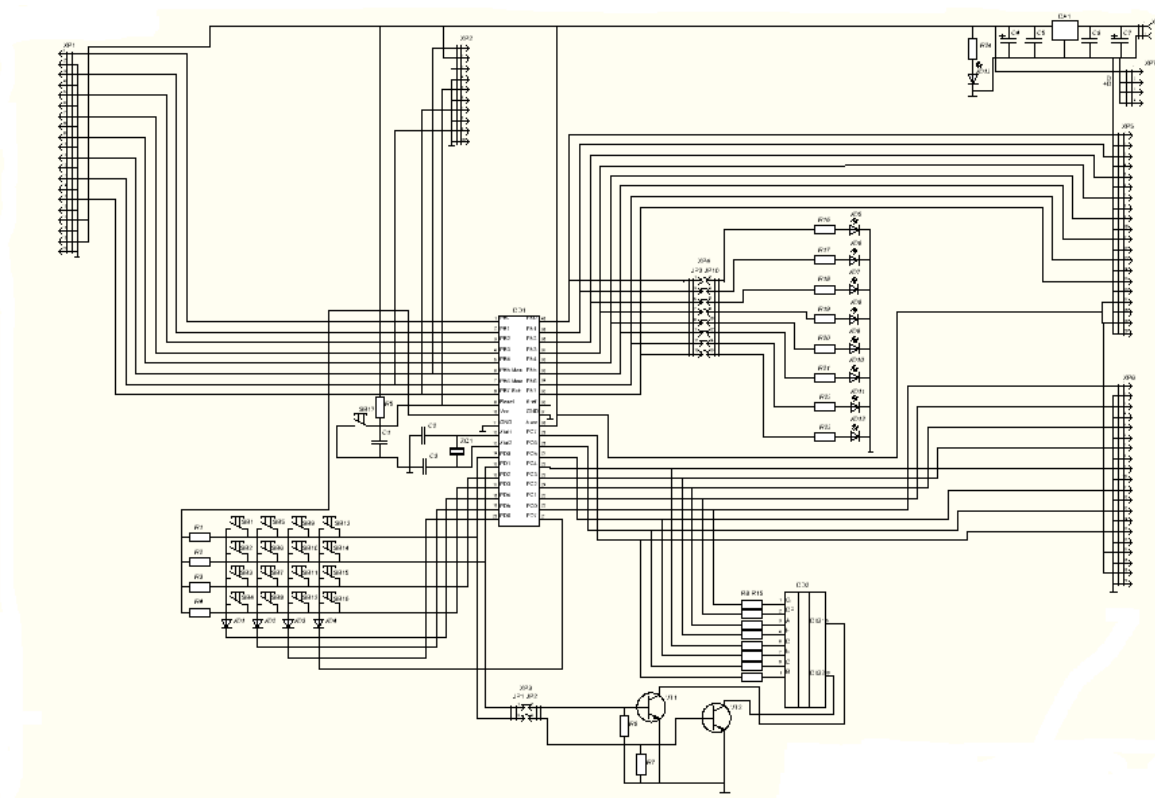


Рисунок 3.1. Схема електрична принципова.

Схема електрична принципова (рисунок 3.1) та специфікація елементів розробленого стенду для дослідження мікроконтролера сімейства AVR ATmega32A міститься у додатку Б та В відповідно.

Кожен елемент, що входить у склад виробу і зображений на схемі, має літерно-цифрове позиційне позначення. Цифри порядкових номерів елементів і їх літерні і позиційні позначення виконуємо одним розміром шрифту по можливості з правого боку або над ними. Порядкові номери елементів на схемі записуємо, як правило, зверху вниз в напрямі зліва направо [18].

Конструкторську документацію необхідну для виготовлення стенду виконуємо за допомогою програм Splan та Sprint–Layout.

					БР.171.061.001 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		50

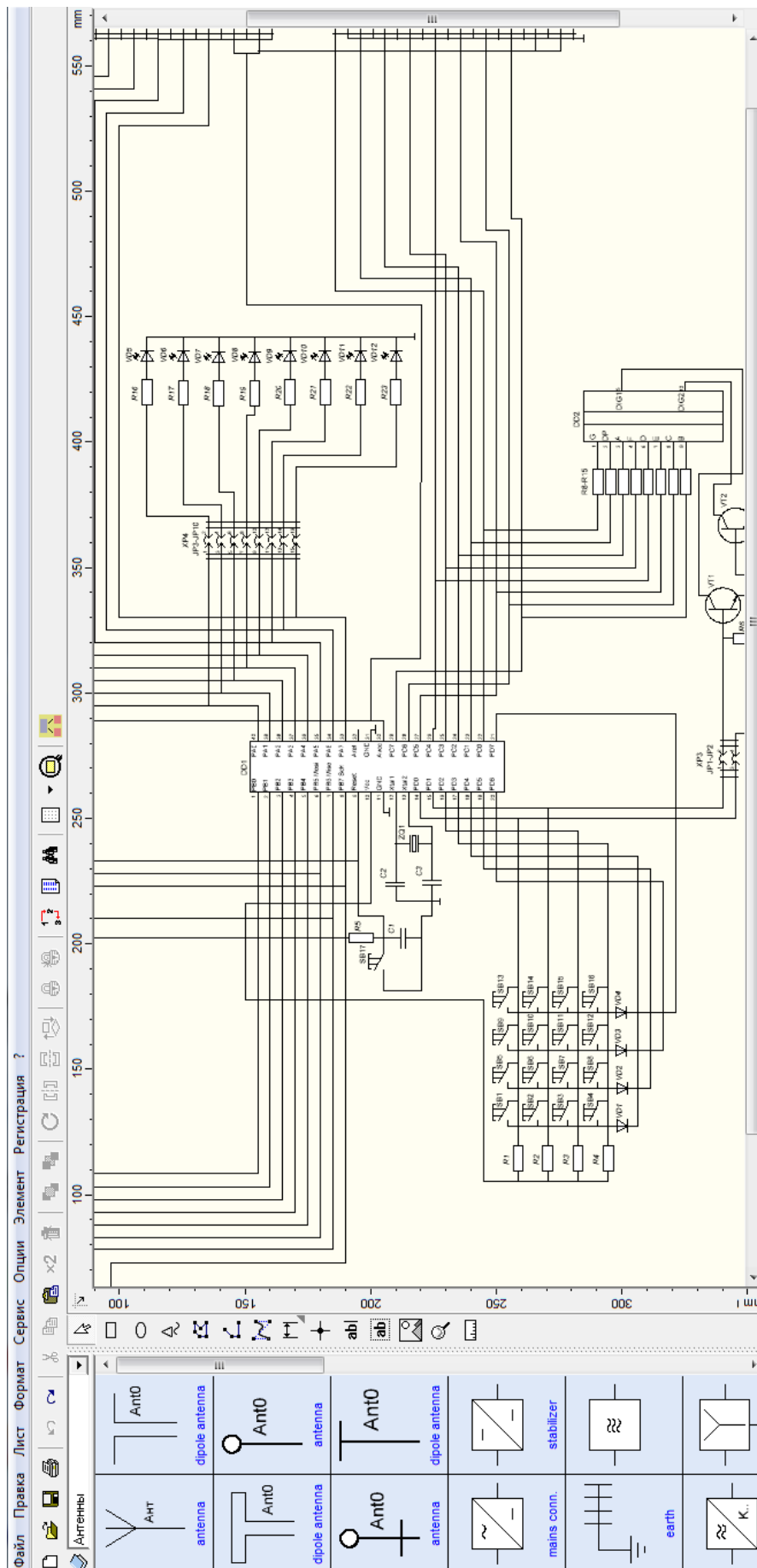


Рисунок 3.2. Видяд головного вiкна програмного забезпечення Sprint

Програмне середовище Splan – простий і зручний інструмент для креслення електронних і електричних схем (рисунок 3.2), дозволяє легко переносити стандартні позначення електро-радіо елементів (ЕРЕ) та символи з бібліотеки елементів на схему і прив'язувати їх до координатної сітки. У Splan є багато інструментів для креслення і редагування, які роблять розробку схем зручною і ефективною, такі як автонумерація елементів, складання переліку елементів та інші.

Бібліотека відмінно організована, елементи розділені на відповідні групи, для вставки елементу схеми з бібліотеки, досить просто його перетягнути на лист. Є можливість зробити власну бібліотеку елементів, якщо потрібного елементу раптом невиявилось.

При розробці друкованої плати використовуємо програму Sprint–Layout.

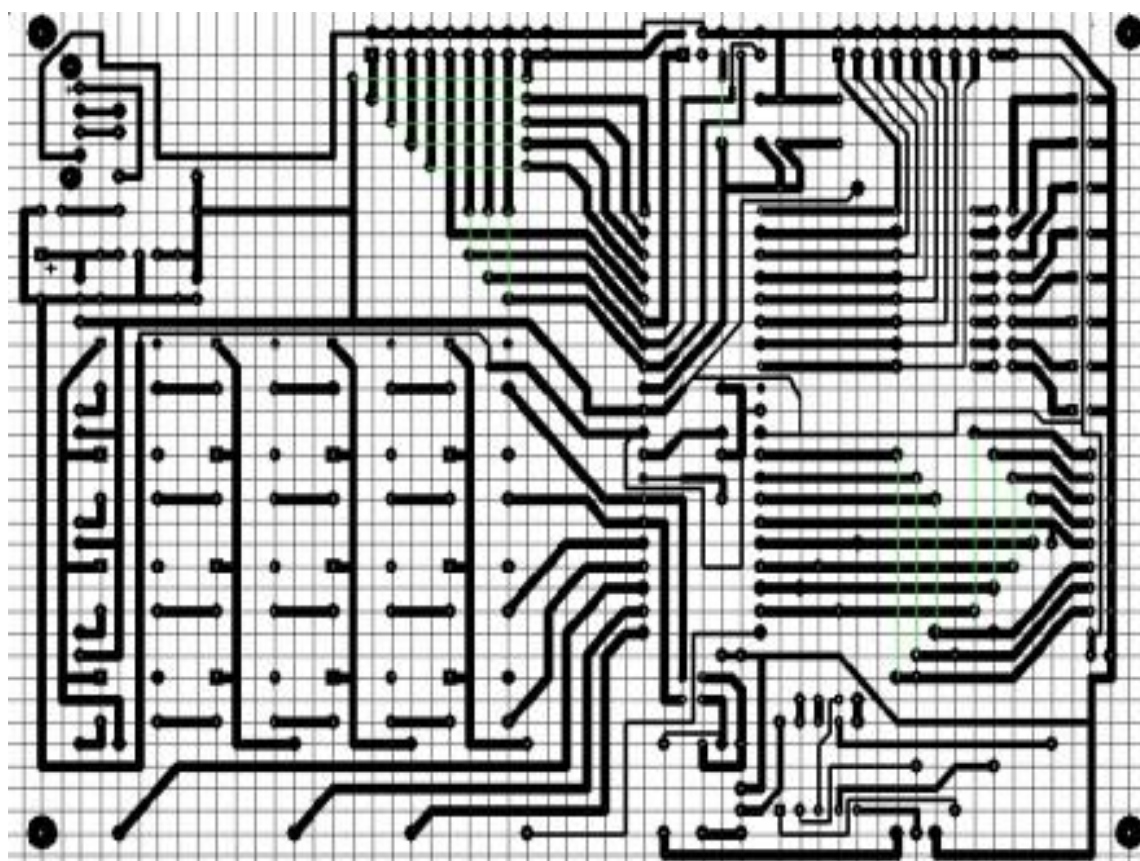


Рисунок 3.3. Креслення друкованої плати.

Креслення друкованої плати стенду (рисунок 3.3) міститься у додатку Г.

Sprint–Layout - це просте і дуже зручне програмне забезпечення для розробки одно- та двосторонніх друкованих плат, розміри яких не перевищують 300×300 мм.

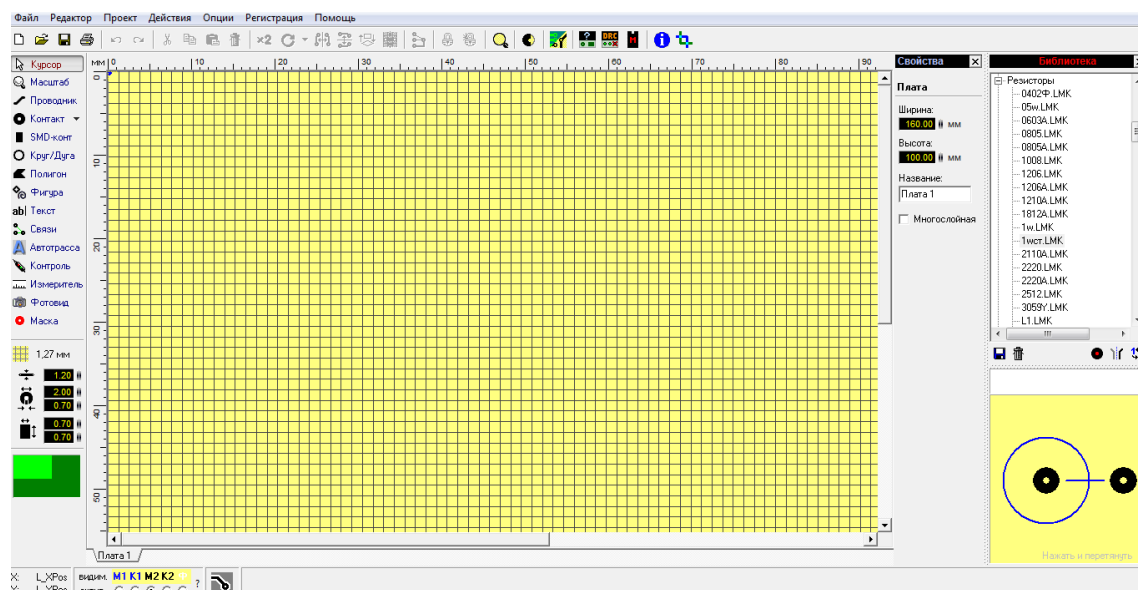


Рисунок 3.4 .Вигляд головного вікна програмного забезпечення Sprint–Layout

Програма Sprint–Layout містить інструменти для малювання друкованих провідників, многогранників, написів і т.д.; лупу для вибору зручного масштабу перегляду креслення, а також має бібліотеку компонентів макросів(рисунок 3.4).

Розробку креслення друкованої плати починаємо з нанесення координатної сітки. За основний крок координатної сітки приймається 2,54 мм [19].

Креслення друкованої плати – містить всі відомості, необхідні для її виготовлення і контролю: зображення друкованої плати з боку друкованого монтажу; розміри, граничні відхилення всіх її елементів (отворів, провідників), а також розміри відстаней між ними; необхідні технічні вимоги, відомості про матеріал.

Після створення схеми електричної принципової, переліку елементів та креслення друкованої плати є можливість вивести документи на друк. Для цього в програмах Splan та Sprint–Layout передбачені всі необхідні налаштування, такі як:

- попередній перегляд;

- вибір формату листа для друку;
- зміна масштабу;
- орієнтація листа;
- та інші.

					БР.171.061.001 ПЗ	Арк.
						54
Змн.	Арк.	№ докум.	Підпис	Дата		

Висновки до розділу 3

Була розроблена конструкторсько-технічна документація з використанням спеціалізованого програмного забезпечення

Наведений детальний опис спеціалізованого програмного забезпечення для створення конструкторсько-технічної документації.

					БР.171.061.001 ПЗ	Арк.
						55
Змн.	Арк.	№ докум.	Підпис	Дата		

4 ЕКСПЕРИМЕНТАЛЬНА ЧАСТИНА: Технологічний процес виготовлення плати електронного стенду на основі мікроконтролера для налагодження програм.

Після завершення розробки конструкторсько-технічної документації (описано в п. 3.1 та 3.2) виконується технологічний процес виготовлення плати. Цей процес включає в себе виготовлення плати, компоновку стенду, пайку елементів, та по завершенню, безпосередньо налаштування і перевірку роботи стенду.

Процес виготовлення плати не є складним, так як включає в себе стандартну технологію. На початку на підготовану пластину фольгованого стеклотекстоліту (можна також використовувати фольгований гетинакс) наноситься малюнок креслення друкованої плати. Це можна виконати як шляхом теплового перенесення креслення надрукованого на спеціальному глянцевого папері за допомогою лазерного принтера, так і дорогим методом – за допомогою використання фоторезисту. Після того як на мідну фольгу нанесено малюнок доріжок, виконується травлення плати. Для травлення плати можна застосовувати різні методи на різні (спеціальні) травники. Дана плата була витравлена за допомогою агресивного кислотного середовища (розчин хлорного заліза). Також на даному етапі виготовлення можуть застосовуватися фрезерні станки, електро-хімічний метод, та ін [20].

Після повного стравлення непокритих тонером ділянок плати (так званого – формування провідних доріжок) плата промивається та сушиться. За тим йде процес лудіння. Лудіння теж може здійснюватися декількома методами. Найдоцільніше використовувати сплав Розе або сплав Вуда, але сплав Вуда володіє високою токсичністю.

Сплав Розе олов'яно-свинцево-вісмутовий сплав який має високу адгезію та низьку температуру плавлення, що забезпечує покриття доріжок тонким шаром сплаву при відносно не високих температурах (80-100 °C).

					БР.171.061.001 ПЗ	Арк.
						56
Змн.	Арк.	№ докум.	Підпис	Дата		

Процес свердління отворів для монтажу компонентів зазвичай виконується після процесу лудіння, але також може виконуватися безпосередньо перед лудінням, так як це не дуже впливає на якість готового виробу. При свердлінні отворів, розмір свердла обирається відповідно до розміру ніжок елемента що буде поміщений в отвір.

Компоновка плати налагоджувального стенду відбувається виключно в повній відповідності схемі електричній принциповій.

Пайка відбувається за допомогою каніфолі (можна використовувати кислоти, але вони знижують час роботи приладу так як з часом пошкоджують ніжки та безпосередньо компоненти), та м'якого олов'яно-свинцевого сплаву, наприклад: ПОС-60, ПОС-40, ПОС-30. Сам процес виконується за допомогою паяльника потужністю 25-40 Вт (не рекомендується використовувати паяльник більшої потужності щоб уникнути перегріву елементів чи розшарування доріжок), або за допомогою паяльної станції будь якої фірми виробника (рисунок 4.1).



Рисунок 4.1. Паяльна станція "4 в 1" Handskit 909D

Програмування та перевірка роботи програм відбувається після повного завершення пайки та при повній комплектації стенду. Це відбувається за допомогою спеціального програматора для AVR мікроконтролерів, та спеціалізованого програмного забезпечення (описано в п. 5.1 та 5.2)

					БР.171.061.001 ПЗ	Арк.
						58
Змн.	Арк.	№ докум.	Підпис	Дата		

Висновки до розділу 4

Покроково розглянуто технологічний маршрут виготовлення стенду. Описано технічні засоби та основні матеріали, що використовувалися в процесі виготовлення.

					БР.171.061.001 ПЗ	Арк.
						59
Змн.	Арк.	№ докум.	Підпис	Дата		

5 РОЗРАХУНКОВА ЧАСТИНА

5.1 Мова програмування Assembler

Асемблер, мова асемблера – це проста мова програмування для програмованої комп'ютерної системи (процесор, мікроконтролер, комп'ютер або інший програмований пристрій), що має сувору відповідність між мовними операторами і машинними інструкціями. Асемблер також називається символічним машинним кодом або мнемонічним кодом.

У найпростішому випадку асемблер переводить речення вихідної програми в об'єкт (команда, константа) модуля навантаження. У цьому випадку відносно розташування об'єктів в модулі навантаження і, нарешті, в пам'яті машини, визначається порядком пропозицій у початковій програмі для автокодів, і цілком залежить від програміста. Асемблер також виконує допоміжні функції, наприклад, підготовку документів для друку необхідної форми, реєстрацію зв'язків цієї програми з іншими програмами тощо.

Для перенесення зазвичай потрібні два перегляди початкової програми: при першому перегляді пам'яті значенням присвоюються символічні імена, при другому перегляді робоча програма розроблена як модуль завантаження. Під час процесу перекладу асемблер виконує повний синтаксичний контроль вихідної програми (див. Програми розбору), забезпечуючи при цьому достатньо точну діагностику помилок позицій і символів.

Команди мови відповідають машинним кодам відповідного процесора або мікроконтролера. Фактично, мова асемблера є зручною символічною формою для запису машинних інструкцій. Як результат, програми, написані для одного типу процесора, будуть працювати на іншому. Мова асемблера також включає засоби для створення міток і переходів, необхідні для створення циклів і вилок. Можливі інструменти для створення макросів і процедур. Кожна сім'я (модель) мікропроцесорів має свій власний набір інструкцій і, відповідно, свій власний набір інструкцій на асемблері [21].

					БР.171.061.001 ПЗ	Арк.
						60
Змн.	Арк.	№ докум.	Підпис	Дата		

5.1.1 Переваги і недоліки

- мінімальна кількість надлишкового коду (з меншою кількістю команд і доступом до пам'яті). У результаті висока швидкість і менший розмір програми;
- великі обсяги коду, велика кількість додаткових невеликих завдань;
- погана читаність коду, проблеми з обслуговуванням (налаштування, додавання функцій)
- складність реалізації парадигм програмування та інших складних конвенцій, складність спільного розвитку;
- наявність меншої кількості бібліотек, сумісність;
- прямий доступ до апаратних засобів: порти вводу / виводу, спеціальні регістри процесорів;
- можливість написання коду, мета програмування, якого немає необхідності програмного інтерпретатора;
- максимальна «придатність» для потрібної платформи (використання спеціальних інструкцій, технічних особливостей «заліза»);
- неможливість роботи на платформах з іншою (несумісною) архітектурою.

5.2 Середовище програмування

AVR Studio - це інтегрована платформа розробки (IDP) для розробки і налагодження всіх додатків мікроконтролерів AVR і SAM. AVR Studio IDP надає вам зручну і просту у використанні середу для написання, складання та налагодження ваших додатків, написаних на C / C ++ або асемблерному коді. Він також легко підключається до відладчик, програмістам і наборам для розробки, що підтримує пристрої AVR і SAM (рисунок 5.1).

					БР.171.061.001 ПЗ	Арк.
						61
Змн.	Арк.	№ докум.	Підпис	Дата		

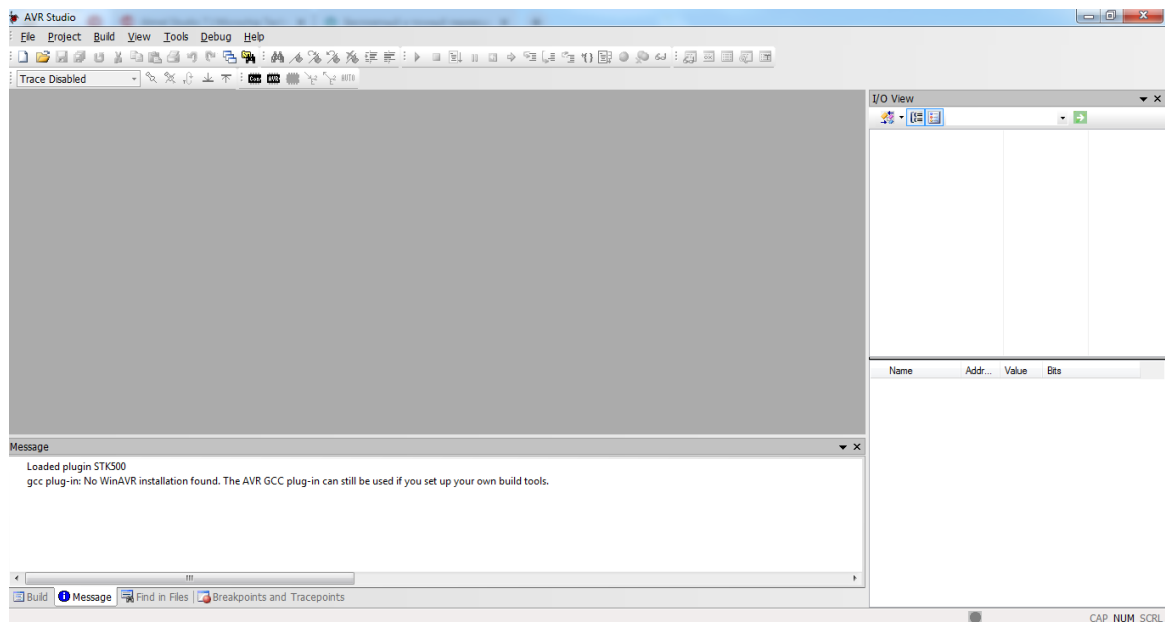


Рисунок 5.1. Вигляд головного вікна програмного забезпечення AVR Studio

Крім того, AVR Studio включає в себе Atmel Gallery, онлайн-магазин додатків, який дозволяє розширити середовище розробки за допомогою плагінів, розроблених Microchip, а також сторонніми постачальниками інструментів і програмно-апаратних засобів. AVR Studio також може легко імпортувати ваші ескізи Arduino як проекти C ++, забезпечуючи простий шлях переходу з Makerspace в Marketplace.

Ключова особливість AVR Studio:

- Підтримка більше 500 пристроїв AVR і SAM
- Велика бібліотека вихідного коду, включаючи драйвери, стеки зв'язку, понад 1600 прикладів проектів з вихідним кодом, графічні сервіси і сенсорні функції за допомогою Advanced Software Framework (ASF)
- Розширення IDE через Atmel Gallery, інтернет-магазин додатків, для інструментів розробки та програмно-апаратних засобів від Microchip і третіх осіб
- За допомогою Atmel QTouch Composer можна налаштовувати ємнісні сенсорні рішення, перевіряти продуктивність системи, контролювати енергоспоживання і дані і графіки трасування в режимі реального часу.

					БР.171.061.001 ПЗ	Арк.
						62
Змн.	Арк.	№ докум.	Підпис	Дата		

- Налаштуйте і протестуйте продуктивність бездротових проектів з Wireless Composer, які працюють на цільовому пристрої.
- Написання і налагодження C / C ++ і асемблера за допомогою вбудованого компілятора
- Розширені функції налагодження містять складні точки переривання даних, підтримку неінтрузивного трасування (пристрої SAM3 і SAM4), статистичне профілювання коду, трасування / моніторинг переривань, трасування опитуваних даних (пристрої Cortex-M0 +), відстеження змінних в реальному часі з додатковою відміткою часу.
- Інтегрований редактор з візуальним супроводом
- Майстер проектів, що дозволяє створювати проекти з нуля або з великої бібліотеки прикладів дизайну
- Внутрисистемне програмування і налагодження забезпечують інтерфейс для всіх внутрісхемних програмістів і отладчиков Atmel.
- Створення прозорих налагоджувальних уявлень в ЦП і периферійних пристроях для простої розробки і налагодження коду
- Симуляція повного чіпа для точної моделі процесора, переривань, периферійних пристроїв і зовнішніх подразників

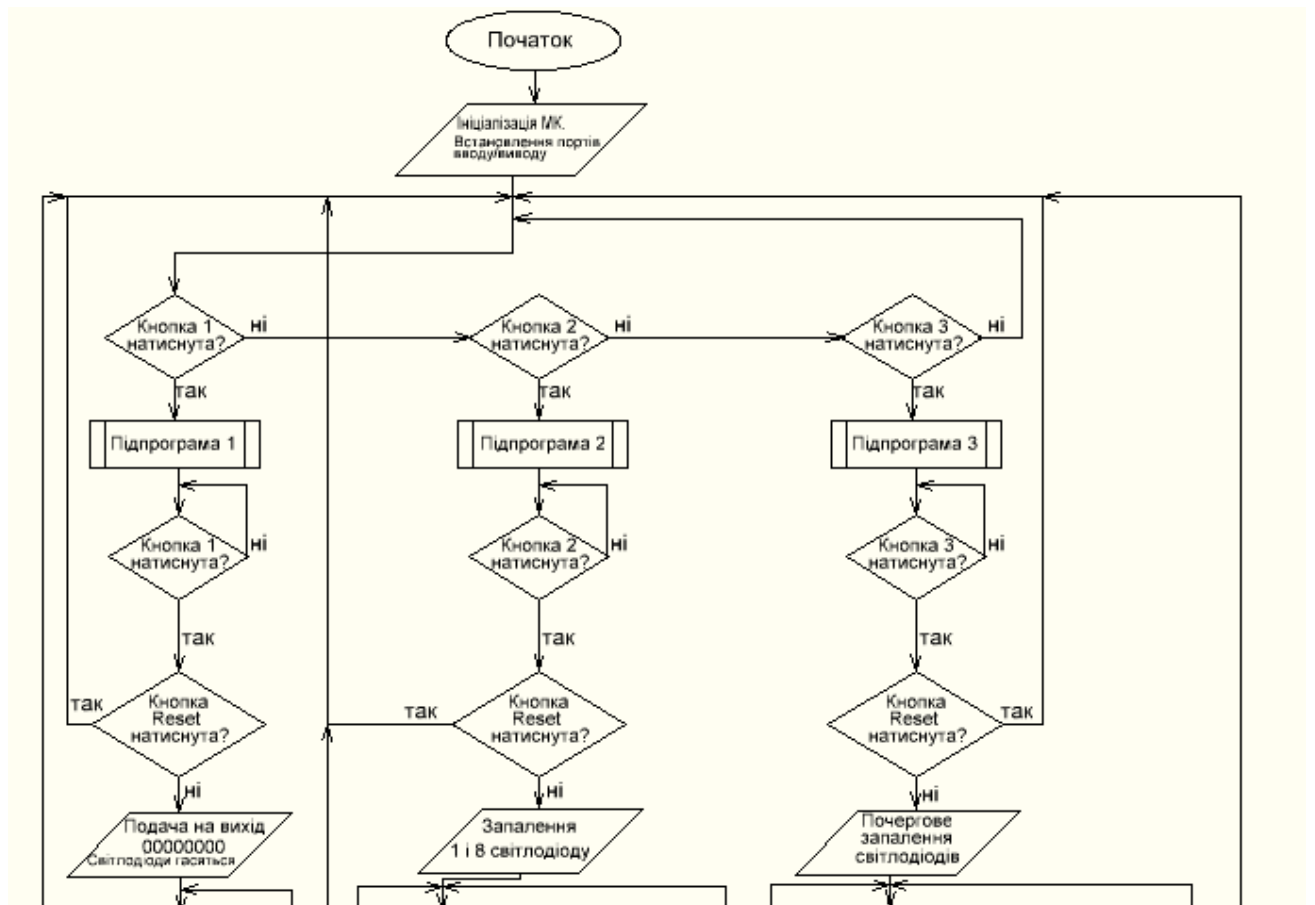
5.3 Опис розробленого алгоритму базової програми електронного стенду

Метою написання програми є перевірка коректності роботи електронного стенду та перевірка роботи компонентів стенду. За допомогою даної програми можна перевірити коректність роботи блоку світлодіодної індикації, та роботу порту для підключення периферії. Також цим методом можна перевірити роботу інших комплектуючих. Така перевірка потрібна для забезпечення точної роботи стенду при налагодженні більш складних програм або при використанні стенду для управління периферійними пристроями. Лістинг програми міститься в Додатку Д

До особливостей програми можна віднести:

					БР.171.061.001 ПЗ	Арк.
						63
Змн.	Арк.	№ докум.	Підпис	Дата		

- програма складається з трьох підпрограм, кожна з яких працює з різними режимами роботи світлодіодів: статична індикація, покрокова зміна комбінації, динамічна індикація (з невеликою швидкістю).
- перед виконанням наступного кроку кожної програми задане неперервне опитування кнопки для покрокового виконання програми, це забезпечує можливість спостерігати кожен момент зміни стану світлодіодів.
- в програмі закладений незалежне безперервне опитування кнопки «RESET» що забезпечує можливість зупинки виконання програми/перемикання між програмами в будь-який зручний момент.



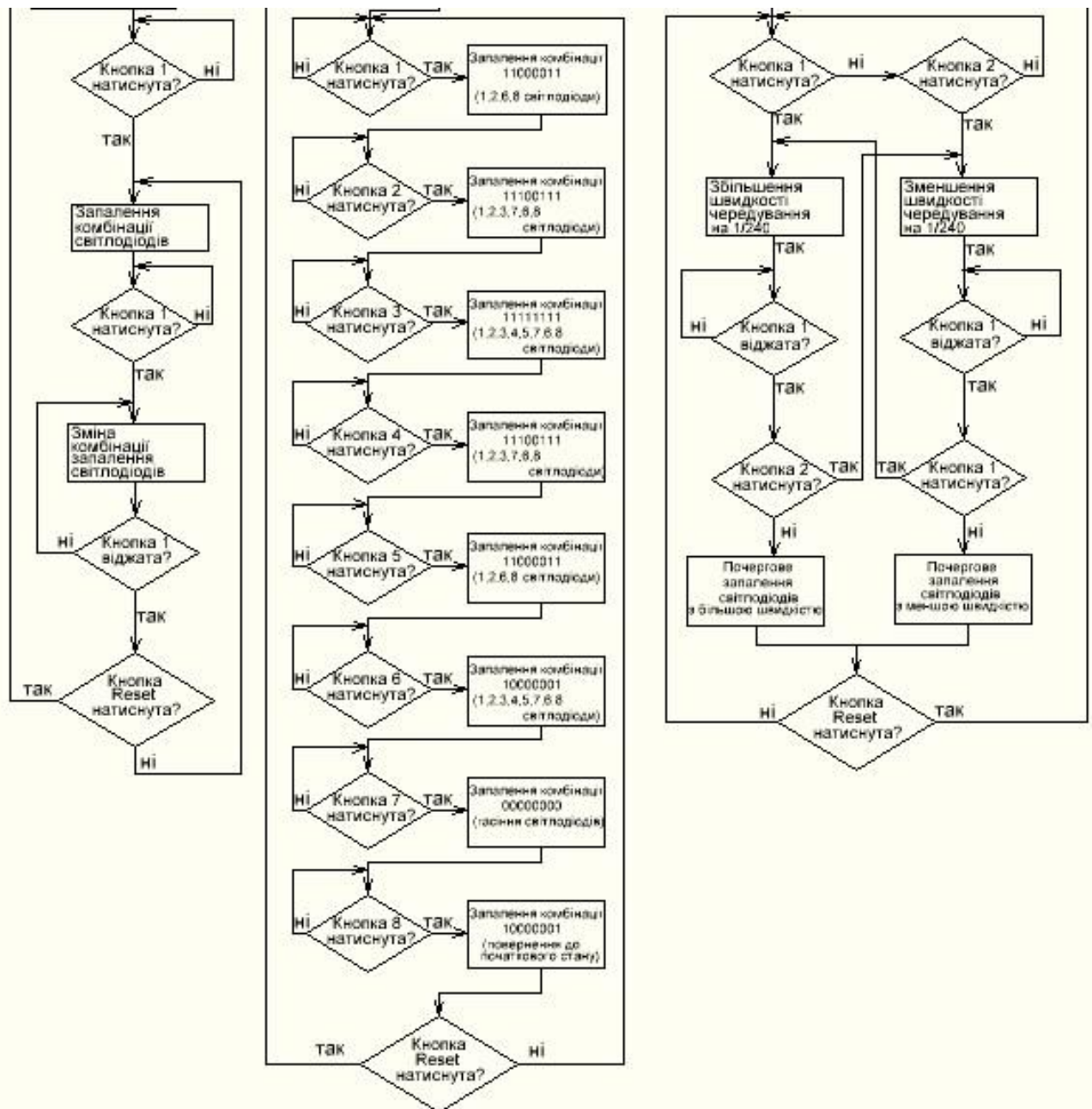


Рисунок 5.2. Алгоритм програми

Висновки до розділу 5

Наведено основні особливості мови асемблер. Розглянуто основні аспекти програмного забезпечення AVR Studio.

Наведено розроблений в дипломному проекті алгоритм програми для перевірки коректності роботи електронного стенду. Алгоритм є простим. Програма забезпечує можливість перевірки функціонування блоків стенду. За допомогою попереднього програмування стенду можна перевірити коректність роботи блоків і окремих елементів стенду.

					БР.171.061.001 ПЗ	Арк.
						66
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

При огляді науково-технічної літератури було проаналізовано принцип роботи мікропроцесора та мікроконтролера, відмінності між мікропроцесором і мікроконтролером, різновиди мікропроцесорів компанії Intel, типи мікроконтролерів.

Особливу увагу приділено: значенню розмірів процесу процесора, складовим компонентам мікропроцесорів, значенню регістрів спеціальних функцій.

Проаналізовано основні характеристики та переваги мікроконтролера ATmega32A, який використовувався як основа електронного стенду при виконанні дипломного проекту

Зроблено огляд існуючих прототипів електронних стендів для налагодження роботи та візуальної перевірки роботи програм мікроконтролерів.

Розроблений стенд для налагодження програм має переваги у порівнянні з заводським аналогом так як комплектація стенду підбирається під виконання визначених завдань, а в стендах які виготовляються спеціалізованими компаніями комплектація є визначеною і зазвичай зміні чи доопрацюванню не підлягає.

Розроблений стенд на основі мікроконтролера для налагодження програм також має такі переваги:

- зручність виконання лабораторних робіт, які пов'язані із програмуванням мікроконтролерів сімейства AVR;
- можливість удосконалення;
- низький рівень енергоспоживання;
- можливість виконання в одному комплексі декількох задач;
- низька собівартість.

Наведений детальний опис будови стенду, роботи функціональних частин та призначення окремих елементів.

					БР.171.061.001 ПЗ	Арк.
						67
Змн.	Арк.	№ докум.	Підпис	Дата		

Була розроблена конструкторсько-технічна документація з використанням спеціалізованого програмного забезпечення

Наведений детальний опис спеціалізованого програмного забезпечення для створення конструкторсько-технічної документації.

Покроково розглянуто технологічний маршрут виготовлення стенду. Описано технічні засоби та основні матеріали, що використовувалися в процесі виготовлення.

Наведено основні особливості мови асемблер. Розглянуто основні аспекти програмного забезпечення AVR Studio.

Наведено розроблений в дипломному проекті алгоритм програми для перевірки коректності роботи електронного стенду. Алгоритм є простим. Програма забезпечує можливість перевірки функціонування блоків стенду. За допомогою попереднього програмування стенду можна перевірити коректність роботи блоків і окремих елементів стенду.

					БР.171.061.001 ПЗ	Арк.
						68
Змн.	Арк.	№ докум.	Підпис	Дата		

ПЕРЕЛІК ВИКОРИСТАНОЇ НАУКОВО-ТЕХНІЧНОЇ ЛІТЕРАТУРИ

1. Якименко Ю.І. Мікропроцесорна техніка: підручник/ Якименко Ю. І., Терещенко Т.О., Сокол Є. І.– К.: Кондор, 2004. – 440с. (дата звернення: 02.10.2018)
2. Microprocessors: the engines of the digital age [Електронний ресурс] – Режим доступу: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5378251/> // (дата звернення: 02.10.2018)
3. Microprocessor [Електронний ресурс] – Режим доступу: <https://en.wikipedia.org/wiki/Microprocessor> // (дата звернення: 02.10.2018)
4. How the microprocessor works [Електронний ресурс] – Режим доступу: <http://wisemotors.ru/uk/zhelezo/how-it-works-microprocess...nd-to-the-processor/> // (дата звернення: 11. 10.2018)
5. MTE Explains: What Is a Processor's Process Size and Why Does It Matter? [Електронний ресурс] – Режим доступу: <https://www.maketecheasier.com/processors-process-size/> // (дата звернення: 11.10.2018)
6. A beginner's guide to microcontrollers [Електронний ресурс] – Режим доступу: <https://www.microcontrollertips.com/a-beginners-guide-to-microcontrollers-faq/> // (дата звернення: 11. 10.2018)
7. What is a Microcontroller? Where they are used? [Електронний ресурс] – Режим доступу: <https://circuitdigest.com/article/what-is-a-microcontroller> // (дата звернення: 20. 10.2018)
8. What are microcontrollers used for? [Електронний ресурс] – Режим доступу: <https://www.quora.com/What-are-microcontrollers-used-for> // (дата звернення: 21. 10.2018)
9. Know About Difference Between Microcontroller and Microprocessor [Електронний ресурс] – Режим доступу: <https://www.tektron.com/know-difference-microcontroller-microprocessor/> // (дата звернення: 25. 10.2018)

					БР.171.061.001 ПЗ	Арк.
						69
Змн.	Арк.	№ докум.	Підпис	Дата		

10. What is the difference between a microprocessor and microcontroller?
[Електронний ресурс] – Режим доступу: <https://www.quora.com/What-is-the-difference-between-a-microprocessor-and-microcontroller> // (дата звернення: 11.11.2018)
11. Types of AVR Microcontroller – Atmega32 [Електронний ресурс] – Режим доступу: <https://www.elprocus.com/types-of-avr-microcontroller-atmega32-and-atmega8/> // (дата звернення: 11.11.2018)
12. Микроконтроллер серии АТМЕГА [Електронний ресурс] – Режим доступу: <http://www.kosmodrom.com.ua/el.php?name=%20ATMega32A-PU> // (дата звернення: 20.11.2018)
13. Зубчук В.И. Справочное пособие по цифровой схемотехнике: справочник/ Зубчук В.И., Сигорский В. П., Шкуро А.Н. – К.: Техника, 1990. – 448 с. // (дата звернення: 21.11.2018)
14. Корнеев В.В. Современные микропроцессоры / Корнеев В.В. – СПб.: БХВ-Петербург, 2003. – 448 с. // (дата звернення: : 21.11.2018)
15. Локазюк В. М. Мікропроцесори та мікроЕОМ у виробничих системах. Навчальний посібник для вузів/ Локазюк В. М.–Хмельницький: ТУП, 2001. – 244 с. // (дата звернення: 02.12.2018)
16. Голубцев М.С. Микроконтроллеры AVR: от простого к сложному/ Голубцев М.С. – М.: Солон-Пресс, 2003. – 288 с. // (дата звернення: 02.12.2018)
17. Ревич Ю.В. Практическое программирование микроконтроллеров Atmel AVR на языке ассемблера/ Ревич Ю. В.. – СПб.: БХВ-Петербург, 2008. – 384 с. // (дата звернення: 02.03.2019)
18. Угрюмов Е.П. Цифровая схемотехника/ Угрюмов. Е.П. – СПб.: БХВ-Петербург, 2001. – 528с. // (дата звернення: 20.03.2019)
19. Sprint-Layout [Електронний ресурс] – Режим доступу: http://cxem.net/software/sprint_layout.php // (дата звернення: 20.04.2019)
20. МЕТОДЫ ИЗГОТОВЛЕНИЯ ПЕЧАТНЫХ ПЛАТ [Електронний ресурс] – Режим доступу: <https://pcbdesigner.ru/pcb/sposobi-izgotovleniya->

					БР.171.061.001 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		70

pechatnih-plat/metody-izgotovleniya-pechatnyx-plat.html // (дата звернення:
10.05.2019)

21. Мова асемблера [*Електронний ресурс*] – Режим доступу:
https://howlingpixel.com/i-uk/Мова_асемблера // (дата звернення: 25.05.2019)

					БР.171.061.001 ПЗ	Арк.
						71
Змн.	Арк.	№ докум.	Підпис	Дата		

ДОДАТКИ

ДОДАТОК А
Технічне завдання

ТЕХНІЧНЕ ЗАВДАННЯ

1. Найменування і область використання.

Електронний стенд на основі мікроконтролера для налагодження програм. Орієнтований на мікропроцесорної техніки.

2. Основа для виконання роботи.

Основою для проектування є завдання на дипломний проект.

3. Мета і призначення розробки.

Метою розробки є схемотехнічне проектування апаратно – програмного комплексу електронного стенду на основі мікроконтролера для налагодження програм. Стенд призначений для налагодження програм та перевірки коректності їх роботи.

4. Технічні вимоги.

4.1. Вимоги до функціональних характеристик системи.

- Можливість забезпечувати живлення від блоку живлення та USB;
- Можливість перепрограмування мікроконтролера без потреби демонтажу та зняття з плати стенду;
- Наявність вільних портів (штекерних роз'ємів) для підключення периферійних пристроїв;
- Можливість завершити програму на будь-якому етапі виконання.

4.2. Вимоги до надійності системи.

Час напрацювання за раптовими відмовами не менше 20 тис. годин.

4.3. Вимоги до технологічності.

Використання сучасної елементної бази з покращеними експлуатаційними характеристиками. Можливість виготовлення на обладнанні, до якого не висувається підвищених вимог.

					БР.171.061.001 ТЗ					
Змн.	Арк.	№ докум.	Підпис	Дата						
Розроб.		Аршан Є.В.			Розробка електронного стенду на основі мікроконтролера для налагодження програм. Технологічне завдання		Літ.	Арк.	Аркушів	
Перевір.		Семікіна Т.В.							1	2
Реценз.		Бобренко Ю.М.					«КПІ ім. Ігоря Сікорського», ФЕЛ, ЕПП, гр. ДЕ-п61			
Н. Контр.		Чадюк В.О.								
Затверд.		Писаренко Л.Д.								

4.4. Вимоги до рівня уніфікації та стандартизації.

Використання технологій та елементів, регламентованих стандартами, прийнятими на території України. Максимально уніфікувати деталі.

5. Вимоги до складових частин виробу, сировини та експлуатаційних матеріалів.

Використовувати матеріали та компоненти серійного виробництва помірної вартості. Конструкція виробу має бути ремонтпридатною та забезпечувати можливість заміни елементів.

6. Результати роботи.

Робота повинна містити наступні документи:

- пояснювальну записку;
- схему електричну принципову;
- креслення друкованої плати;
- складальне креслення друкованої плати;
- перелік елементів;
- специфікацію;
- додатки.

					БР.171.061.001 ТЗ	Арк.
						2
Змн.	Арк.	№ докум.	Підпис	Дата		

ДОДАТОК Б

Схема електрична принципова електронного стенду
на основі мікроконтролера для налагодження програм.

ДОДАТОК В

Специфікація електронного стенду
на основі мікроконтролера для налагодження програм.

Зона	Поз.	Позначення	Найменування	Кіл.	Прим.		
			Деталі				
	1	ДП.171.061.009.01 Е4	Друкована плата	1			
			Конденсатори				
	2		ST4-0,1-50В ОЖО 464.111 ТУ	3	C1, C5,C6		
	3		CC4-20-50В ОЖО 464.111 ТУ	2	C2,C3		
	4		CL-33-25В ОЖО 464.111 ТУ	1	C4		
	5		CL-100-35В ОЖО 464.111 ТУ	1	C7		
	6		Стабілізатор 78L05	1	DA1		
	7		Мікроконтролер АТmega32А	1	DD1		
	8		Індикатор 2 розряда SN 420362N/16	1	DD2		
	9		Джампер JM-G	10	JP1- JP10		
			Резистори		R1-R7		
	10		МЛТ 10к ±10% ОЖО 467.104 ТУ	7	R1-R7		
	11		CFR 360 ±10% ОЖО 467.104 ТУ	17	R8-R24		
	12		Кнопка тактова KFC-A06-5В	17	SB1- SB17		
			БР.171.061.001 ПЕ				
Зм.	Арк	№ докум.	Підп.	Дата			
Розроб.	Аршан Є.В.				Електронний стенд на основі мікроконтролера для налагодження програм. Специфікація		
Перевір.	Семікінв Т.В						
Н.контр	Надюк В.О						
Затверд.	Ісаренко Л.Д.						
					Літ.	Аркуш	Аркушів
						1	2
					«КПІ ім. І.Сікорського», ФЕЛ, ЕПП, гр. ДЕ-пб1		

[illegible]

ДОДАТОК Г

Креслення друкованої плати електронного стенду
на основі мікроконтролера для налагодження програм.

ДОДАТОК Д

Лістинг програми електронного стенду
на основі мікроконтролера для налагодження програм.

ЛІСТИНГ ПРОГРАМИ

Програмний код для електронного стенду

```
.list
.include "C:\Program Files (x86)\Atmel\AVR
Tools\AvrAssembler2\Appnotes\m16def.inc"
.nolist
.def    temp =r16
rjmp    Init      ; Перша виконувана команда
Init:
ser      temp      ; PA, PC, PD - виходи
out      DDRC,temp
out      DDRC,temp
out      DDRD,temp
ldi      temp,0b00000000 ; PB0,7 - входи, останні не використовуються
out      DDRB,temp
ldi      temp,0b11111111 ; PB0,7 - встановлюємо підтяжку
out      PortB,temp
Start_Opros:
sbis     PinB,0
rjmp     Programma_1
sbis     PinB,1
rjmp     Programma_2
sbis     PinB,2
rjmp     Programma_3
rjmp     Start_Opros
Programma_1:
.device  m16def
.nolist
.include "C:\Program Files (x86)\Atmel\AVR
Tools\AvrAssembler2\Appnotes\m16def.inc"
.list
```

					БР.171.061.001 СП	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		84

```

.def temp = r16
rjmp Init_Prog_1 ;
Init_Prog_1:
ser temp
out DDRA, temp ; Зробили порт D на вихід
ldi temp, 0b00000000
out DDRB, temp ; Зробили порт B на вхід
ldi temp, 0b11111111
out PortB, temp ; Встановили підтягуючий резистор на всіх входах порта B
clr temp
out PortA, temp ; Обнулили значення на порті B
Start_Prog1:
sbic PinB, 0 ; перевіряє, чи приходить одиниця на старший біт регістра D
ldi temp, 0b01010101 ; Якщо попередня умова виконується, то заносимо в
робочий регістр в молодший біт одиницю
out PortA, temp ;
sbis PinB, 0 ; перевіряє, чи приходить нуль на старший біт регістра D
ldi temp, 0b10101010 ; якщо попередня умова виконується, то заносимо в
робочий регістр в молодший біт нуль
out PortA, temp ;виводимо в порті B значення
rjmp Start_Prog1 ; зациклюємо програму
Programma_2:
.device m16def
.nolist
.include "C:\Program Files (x86)\Atmel\AVR
Tools\AvrAssembler2\Appnotes\m16def.inc"
.list
.def temp = r16
rjmp Init_Prog2 ;
Init_Prog2:
ser temp
out DDRA, temp ; Зробили порт D на вихід

```

					БР.171.061.001 СП	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		85

```

ldi temp, 0b00000000
out DDRB, temp    ; Зробили порт В на вхід
ldi temp, 0b11111111
out PortB, temp    ; Встановили підтягуючий резистор на всіх входах порта В
ldi temp, 0b00000000
out PortA, temp    ; Обнулили значення на порті В
Start_Prog2:
a:
ldi temp, 0b10000001 ; Якщо попередня умова виконується, то заносимо в
робочий регістр в молодший біт одиницю
out PortA, temp ;
sbic PinB, 0    ; перевіряє, чи приходить одиниця на старший біт регістра D
rjmp a
b:
ldi temp, 0b11000011 ; Якщо попередня умова виконується, то заносимо в
робочий регістр в молодший біт одиницю
out PortA, temp ;
sbic PinB, 1
rjmp b
c:
ldi temp, 0b11100111 ; Якщо попередня умова виконується, то заносимо в
робочий регістр в молодший біт одиницю
out PortA, temp ;
sbic PinB, 2
rjmp c
d:
ldi temp, 0b11111111 ; Якщо попередня умова виконується, то заносимо в
робочий регістр в молодший біт одиницю
out PortA, temp ;
sbic PinB, 3
rjmp d
e:

```

					БР.171.061.001 СП	Арк.
						86
Змн.	Арк.	№ докум.	Підпис	Дата		

ldi temp, 0b11100111 ; Якщо попередня умова виконується, то заносимо в
робочий регістр в молодший біт одиницю

out PortA, temp ;

sbic PinB, 4

rjmp e

f:

ldi temp, 0b11000011 ; Якщо попередня умова виконується, то заносимо в
робочий регістр в молодший біт одиницю

out PortA, temp ;

sbic PinB, 5

rjmp f

g:

ldi temp, 0b10000001 ; Якщо попередня умова виконується, то заносимо в
робочий регістр в молодший біт одиницю

out PortA, temp ;

sbic PinB, 6

rjmp g

h:

ldi temp, 0b00000000 ; Якщо попередня умова виконується, то заносимо в
робочий регістр в молодший біт одиницю

out PortA, temp ;

sbic PinB, 7

rjmp h

i:

ldi temp, 0b00011000 ; Якщо попередня умова виконується, то заносимо в
робочий регістр в молодший біт одиницю

out PortA, temp ;

sbic PinB, 6

rjmp i

j:

ldi temp, 0b00111100 ; Якщо попередня умова виконується, то заносимо в
робочий регістр в молодший біт одиницю

					БР.171.061.001 СП	Арк.
						87
Змн.	Арк.	№ докум.	Підпис	Дата		

```

out  PortA, temp ;
sbic  PinB, 5
rjmp  j
k:
ldi   temp, 0b01111110 ; Якщо попередня умова виконується, то заносимо в
робочий регістр в молодший біт одиницю
out  PortA, temp ;
sbic  PinB, 4
rjmp  k
l:
ldi   temp, 0b11111111 ; Якщо попередня умова виконується, то заносимо в
робочий регістр в молодший біт одиницю
out  PortA, temp ;
sbic  PinB, 3
rjmp  l
m:
ldi   temp, 0b11100111 ; Якщо попередня умова виконується, то заносимо в
робочий регістр в молодший біт одиницю
out  PortA, temp ;
sbic  PinB, 2
rjmp  m
o:
ldi   temp, 0b10000001 ; Якщо попередня умова виконується, то заносимо в
робочий регістр в молодший біт одиницю
out  PortA, temp ;
sbic  PinB, 1
rjmp  o
p:
ldi   temp, 0b00000000 ; Якщо попередня умова виконується, то заносимо в
робочий регістр в молодший біт одиницю
out  PortA, temp ;
sbic  PinB, 0

```

```

rjmp p
rjmp Start_Prog2      ; заціклюємо програму
Programma_3:
.list
.include "C:\Program Files (x86)\Atmel\AVR
Tools\AvrAssembler2\Appnotes\m16def.inc"
.nolist
.def temp =r16
.def Mark240 =r17
.def Counter =r18
.def Speed =r19
rjmp Init_Prog3      ; Перша виконувана команда
Init_Prog3:
ser temp            ; PA0-7 - виходи
out DDRC,temp
out DDRD,temp
ser temp
out DDRA,temp
ldi temp,0b00000000 ; PD0,1 - входи, останні не використовуються
out DDRB,temp      ;
ldi temp,0b00000001 ; При старті включений тільки PA0
out PortA,temp
ldi temp,0b00000011 ; PD0,1 - підтяжка,
out PortB,temp
ldi temp,0b00000101
out TCCR0,temp     ; Частота таймера Ськ/1024 out TCCR0,temp
ldi Mark240,240
ldi Counter,10
ldi Speed,10
Start_Prog3:
sbis PinB,0        ; Перевіряємо кнопку зменшення швидкості
rjmp Start_Prog3    ; Не натиснута, переходиться

```

					БР.171.061.001 СП	Арк.
						89
Змн.	Арк.	№ докум.	Підпис	Дата		

```

inc    Speed      ; Зменшуємо швидкість
cpi    Speed,11    ; Speed = 11?
brne   ReleaseDown ; Переходимо до ReleaseDown, якщо немає
dec    Speed      ; Зменшуємо Speed на 1
ReleaseDown:
sbis   PinB,0      ; Чекаємо відпуску кнопки зменшення
rjmp   ReleaseDown
UpTest:
sbic   PinB,1      ; Перевіряємо кнопку збільшення швидкості
rjmp   Timer       ; Не натиснута, переходиться
dec    Speed      ; Збільшуємо швидкість
brne   ReleaseUp   ; Переходимо до Timer, якщо не 0
inc    Speed      ; Збільшуємо Speed на 1
ReleaseUp:
sbis   PinB,1      ; Чекаємо відпуску кнопки збільшення
rjmp   ReleaseUp
Timer:
in     temp,TCNT0   ; Прочитуємо полягання T/сo в temp
cp     temp,Mark240 ; Порівнюємо temp з Mark240
brne   Start_Prog3  ; Якщо не рівно, повертаємося до start
subi   Mark240,-240 ; Додаємо 240 до Mark240
dec    Counter      ; Зменшуємо Counter на 1
brne   Start_prog3  ; Якщо не 0, повертаємося до мітки Start
mov    Counter,Speed ; Скидаємо Counter
in     temp,PortA    ; Прочитуємо поточний стан СИД
lsl    temp         ; Зрушуємо вліво
brcc   PC+2         ; Якщо перенесення не було опускаємо команду
ldi    temp,0b00000001 ; Скидаємо: PA0 - вкл, останні - викл
out    PortA,temp    ; Виводимо в PORTA
rjmp   Start_Prog3   ; Повертаємося до мітки Start

```

					БР.171.061.001 СП	Арк.
						90
Змн.	Арк.	№ докум.	Підпис	Дата		